

EFFECTS OF CONDITIONING ON THE CONVERGENCE OF  
RANDOMIZED OPTIMIZATION ALGORITHMS

A Dissertation

Presented to the Faculty of the Graduate School  
of Cornell University

in Partial Fulfillment of the Requirements for the Degree of  
Doctor of Philosophy

by

Dennis John Leventhal

August 2009

© 2009 Dennis John Leventhal

ALL RIGHTS RESERVED

# EFFECTS OF CONDITIONING ON THE CONVERGENCE OF RANDOMIZED OPTIMIZATION ALGORITHMS

Dennis John Leventhal, Ph.D.

Cornell University 2009

The connection between the conditioning of a problem instance—the sensitivity of a problem instance to perturbations in the input—and the speed of certain iterative algorithms in solving that problem instance is a recurring topic of study in numerical analysis. This dissertation, consisting of three distinct parts, provides a further connection through the framework of randomized optimization algorithms.

In Part I, we explore how randomization can help asymptotic convergence properties of simple, directional search-based optimization methods. Specifically, we develop a randomized, iterative scheme for estimating the Hessian matrix of a twice-differentiable function. Using this estimation technique, we analyze how it can be used to enhance a random directional search method. From there, we proceed to develop a conjugate-directional search method that incorporates estimated Hessian information without requiring direct use of gradients.

In Part II, we turn our focus to randomized variants of two classical algorithms: coordinate descent methods for systems of linear equations and iterated projection methods for systems of linear inequalities. We then demonstrate that, under appropriate randomization schemes, linear rates of convergence can be bounded (in expectation) in terms of natural linear-algebraic conditioning mea-

asures for these problems. By considering conditioning concepts induced by metric regularity and metric subregularity, we then expand upon these results by examining randomized projection algorithms for convex feasibility problems. Extensions to reflection-based algorithms are also discussed.

Observing that convex feasibility problems can be reformulated into the problem of finding a common zero of maximal monotone operators, we proceed by studying the proximal point method in Part III. Specifically, for the problem of finding a zero of a single maximal monotone operator, we show that metric subregularity of that operator is sufficient for linear convergence of the proximal point method, leading to a convergence rate in terms of the conditioning induced by the modulus of subregularity. This result is then generalized—by considering randomized and averaged proximal point methods—to obtain a convergence rate for the problem of finding a common zero of finitely many such operators.

## **BIOGRAPHICAL SKETCH**

Originally born in New York City, Dennis spent the first eighteen years of his life in Old Bridge, New Jersey, graduating from Old Bridge High School in June 2000. From there, he moved to Pittsburgh, Pennsylvania to attend Carnegie Mellon University and obtained his B.S. in Mathematical Sciences in December 2003 while simultaneously developing minor obsession with the Pittsburgh Steelers. After a six month escape from living in the northeastern United States by moving to the coast of central Florida, Dennis arrived in Ithaca, New York to study Operations Research at Cornell University.

Having recently completed his doctoral studies, Dennis became forced to conclude that spending 22 years in school makes for a relatively uninteresting biography.

*Yeah well, that's just, ya know, like, your opinion, man.*

—Jeffrey Lebowski aka The Dude

## ACKNOWLEDGEMENTS

First and foremost, I would like to thank my dissertation advisor, Adrian Lewis, without whom this work would not have been possible. His mathematical insight has immensely broadened my view of Optimization. Further, his patience and encouragement during the development of this work certainly added to the overall experience.

Next, I would like to thank the other members of my dissertation committee, James Renegar and Michael Todd, not only for the interesting mathematical discussions over the years, but also for providing the initial encouragement to come study Operations Research at Cornell in the first place. Additionally, I'd like to thank Shane Henderson both for his interest in my work and for the helpful comments and suggestions along the way. More generally, I'd like to extend thanks to Cornell University's School of Operations Research and Information Engineering and its faculty, students and staff for making this a wonderful place to study.

Additionally, I'd like to thank my parents for their support during this ordeal and, most importantly, listening to me rant on many occasions.

Finally, I'd like to thank the many friends and acquaintances over the years—too many to name individually (though the two of them who may actually read this will undoubtedly take it personally)—for some combination of the support, encouragement or sheer entertainment value they've provided, even though I know most of them would vehemently deny having contributed to the “support” or “encouragement” categories.

## TABLE OF CONTENTS

Biographical Sketch . . . . .	iii
Dedication . . . . .	iv
Acknowledgements . . . . .	v
Table of Contents . . . . .	vi
List of Figures . . . . .	vii
<b>1 Introduction</b>	<b>1</b>
<b>2 Common Notation and Definitions</b>	<b>11</b>
2.1 Introduction . . . . .	11
2.2 Linear Algebra . . . . .	12
2.3 Convex and Variational Analysis . . . . .	14
2.3.1 The Basics . . . . .	14
2.3.2 Metric Regularity and Subregularity . . . . .	17
2.3.3 Geometry and Metric Regularity . . . . .	21
2.4 Linear Convergence . . . . .	23
<b>3 Randomized Hessian Estimation</b>	<b>26</b>
3.1 Introduction . . . . .	26
3.2 Randomized Hessian Estimation . . . . .	30
3.3 Applications to Algorithms . . . . .	36
3.3.1 Random Search, Revisited . . . . .	36
3.3.2 A Conjugate Directions Algorithm . . . . .	38
3.4 Concluding Remarks for Chapter 3 . . . . .	48
<b>4 Randomized Methods for Linear Constraints</b>	<b>50</b>
4.1 Introduction . . . . .	50
4.2 Randomized Coordinate Descent . . . . .	51
4.2.1 The Basic Result: Positive Semi-Definite Systems . . . . .	51
4.2.2 The General Result: Positive Semi-Definite Systems . . . . .	55
4.2.3 General Linear Systems . . . . .	57
4.3 Randomized Iterated Projections . . . . .	59
4.4 Metric Regularity and Local Convergence . . . . .	68
4.5 Reflection Methods . . . . .	73
4.5.1 Linear Constraints . . . . .	75
4.5.2 Convex Constraints . . . . .	78
4.6 Concluding Remarks . . . . .	81
<b>5 Randomized Proximal Point Methods</b>	<b>83</b>
5.1 Introduction . . . . .	83
5.2 Metric Subregularity and Linear Convergence . . . . .	86
5.2.1 The Main Results . . . . .	86
5.2.2 Projection Algorithms: A Special Case . . . . .	93



<b>6 Open Questions and Future Research</b>	<b>96</b>
<b>Bibliography</b>	<b>99</b>

## LIST OF FIGURES

3.1	Random search with Hessian estimates on convex quadratics . .	39
3.2	Conjugate directions algorithm on convex quadratics . . . . .	46
3.3	Conjugate directions algorithm on Rosenbrock function . . . . .	47
3.4	Conjugate directions and Nelder-Mead algorithms . . . . .	48
4.1	Randomized coordinate descent algorithm for least squares problems . . . . .	60
4.2	Randomized alternating projection algorithm for linear inequalities . . . . .	66
4.3	Randomized Reflections of Equality Systems . . . . .	78
4.4	Randomized Reflections for Inequality Systems . . . . .	79

# CHAPTER 1

## INTRODUCTION

The condition number of a problem instance measures the sensitivity of a solution to small perturbations in its input data. For many problems that arise in numerical analysis, there is often a simple relationship between the condition number of a problem instance and the distance to the set of *ill-posed problems*—those problem instances whose condition numbers are infinite [28]. For example, with respect to the problem of inverting a matrix  $A$ , it is known (see [59], for example) that if  $A$  is perturbed to  $A + E$  for sufficiently small  $E$ , then

$$\frac{\|(A + E)^{-1} - A^{-1}\|}{\|A^{-1}\|} \leq \|A^{-1}\| \|E\| + O(\|E\|^2).$$

Thus, a condition measure for this problem may be taken as  $\|A^{-1}\|$ . Associated with this is the classical Eckart-Young theorem found in [37], relating the above condition measure to the distance to ill-posedness.

**Theorem 1.0.1 (Eckart-Young)** *For any non-singular matrix,  $A$ ,*

$$\min_G \{\|G\| : A + G \text{ is singular}\} = \frac{1}{\|A^{-1}\|}.$$

From a computational perspective, a related and important area of study is that of error bounds. Given a subset of a Hilbert space, an error bound is an inequality that bounds the distance from a test vector to the specified subset in terms of some residual function that is typically easy to compute. In that sense, an error bound can be used both as part of a stopping rule during implementation of an algorithm as well as an aide in proving algorithmic convergence. A comprehensive survey of error bounds for a variety of problems arising in optimization can be found in [87].

With regards to the problem of solving a square, nonsingular linear system  $Ax = b$ , one connection between condition measures and error bounds is immediate. Let  $x^*$  be a solution to the system and  $x$  be any other vector. Then

$$\|x - x^*\| = \|A^{-1}A(x - x^*)\| = \|A^{-1}(Ax - b)\| \leq \|A^{-1}\| \|Ax - b\|, \quad (1.0.2)$$

so the distance to the solution set is bounded by a constant multiple of the residual vector,  $\|Ax - b\|$ , and this constant is the inverse of the one that appears in the context of distance to singularity. Practically speaking, knowledge of the error bound—or of the existence of such an error bound—allows one to know that the distance to the solution set is bounded by a multiple of a more easily-computable quantity—in this case, the residual norm.

The frequent appearance of the term  $\|A^{-1}\|$  in the above results is no coincidence. A recurring paradigm in the area of numerical analysis is the near-equivalence between badly posed problems—those problems which are a small perturbation from being ill-posed—and problems for which weak error bounds exist. Further, these two properties are themselves often associated with problems for which iterative algorithms tend to converge slowly. For example, consider the problem of solving a linear system,  $Ax = b$ , where now  $A$  is a positive-definite matrix. Theorem 1.0.1 and Inequality 1.0.2 show that when  $\|A^{-1}\|$  is large, the distance to ill-posedness is small and the natural error bound is weak. Further, the steepest descent algorithm and the conjugate gradient algorithm are known to be linearly convergent (see [1], [48], among others) with rates  $1 - O(\frac{1}{k(A)})$  and  $1 - O(\frac{1}{\sqrt{k(A)}})$ , respectively, where  $k(A) = \|A\| \|A^{-1}\|$  is a scale-invariant condition measure. If we consider the set of problem instances with a fixed value of  $\|A\|$  (by considering an *a priori* rescaling, for example), this shows that these particular iterative algorithms converge more slowly as  $\|A^{-1}\|$  increases.

There is no shortage of literature on the interplay between the ideas of conditioning and error bounds and, of particular interest to the optimization community, algorithmic efficiency. For example, in the pioneering papers by Renegar in [96], [97], [98], a conditioning notion for linear programming was defined directly in an “Eckart-Young style”—in terms of the distance to infeasibility—and it was shown that this condition measure directly governs the speed of interior point algorithms.

Much work exists in other areas of optimization, as well, relating conditioning, error bounds and algorithmic speed. Sample work includes quadratic programming [116], nonlinear programming [112], semidefinite programming [82], stochastic programming [104] and additional examples for linear programming [110], [57], while a broad variety of applications are discussed in [87] and the many references therein. This list is by no means comprehensive, either across areas of study or within the specifically listed areas.

A broad framework is being built in variational analysis for generalizing this paradigm to nonlinear systems. In the spirit of keeping things as sufficiently general as possible, consider a set-valued mapping,  $\Phi : \mathbb{E} \rightrightarrows \mathbb{Y}$ , satisfying  $\Phi(x) \subseteq \mathbb{Y}$  for  $x \in \mathbb{E}$ . An associated problem is that of finding  $x$  such that  $\bar{b} \in \Phi(x)$  for a given vector  $\bar{b}$ . This framework encompasses a variety of problems, including not only ordinary equation solving, but also feasibility problems, variational inequalities and other optimality conditions.

Naturally, any result will ultimately depend on properties of the mapping  $\Phi$  itself; however, this general framework of set-valued mappings allows for the exploration of the “true nature” of conditioning without being encumbered by a specific problem structure. Our interest in regularity properties of  $\Phi$  will focus

around the area of metric regularity, essentially defined by the existence of a local error bound around  $(\bar{x}, \bar{b})$  with  $\bar{b} \in \Phi(\bar{x})$ . This and related properties will be defined more formally in Section 2.3.2. Thorough surveys about error bounds, metric regularity and related properties can be found in [73], [61] and [33].

Metric regularity provides a broad generalization to the conditioning ideas discussed in the context of linear systems. For example, it's certainly curious that the constant appearing in the Eckart-Young Theorem, Theorem 1.0.1, is the reciprocal of the constant in the natural error bound for linear systems, Inequality 1.0.2. In fact, this inverse relationship between error bounds and distance to ill-posedness is substantially more general. As shown in [32] and discussed briefly in Section 2.3.2, under mild assumptions, metric regularity is the condition under which this relationship holds, but for set-valued mappings instead of being limited to linear systems.

Although the nature of metric regularity makes it an interesting topic of study in its own right, further interest in this property is propagated by the implications of metric regularity when studying specific classes of problems or mappings, often leading back to well-studied regularity assumptions for specific problems. Consider a few prominent examples. Given a convex function,  $f$ , it was shown in [5] that metric regularity of the subdifferential mapping,  $\partial f$ , is equivalent to a type of local quadratic growth condition. For a differentiable, convex inequality system

$$g_i(x) \leq 0 \quad \text{for } i = 1, \dots, m, \tag{1.0.3}$$

it was shown in [72] that, under appropriate conditions, metric regularity of the mapping  $\Phi(x) = [g_1(x), \dots, g_m(x)]^T + \mathbb{R}_+^m$  is equivalent to Abadie's constraint qualification as well as being implied by the more prominently studied Slater

([99]) and Mangasarian-Fromowitz ([27]) constraint qualifications. When the constraints in 1.0.3 are restricted to be affine, metric regularity provides a connection between the classic Hoffman error bound from [58] and the distance to infeasibility studied by Renegar in [96], among others. Further, as shown in [32, Thm. 4.8], the framework of metric regularity provides an alternative method for calculating the distance to infeasibility in such a case by appealing to connections with the calculus of coderivatives (see [103], among others).

Given this connection between error bounds and distance to ill-posedness for set-valued mappings, the question remains as to how this property affects the speed of iterative algorithms. The general use of error bounds to understand the convergence of algorithms has been studied by many authors for a variety of applications; one broad approach that encompasses gradient projection methods, coordinate descent methods and proximal point algorithms, among others, can be found in [77] and [75]. The body of work explicitly establishing a connection between metric regularity and algorithmic performance is still in its infancy; however, the papers [4], [70] and [66] are worthy of mention. This, in fact, is the second of the two major themes we address in this dissertation.

Fundamentally, the primary theme of this dissertation involves the introduction of randomization schemes in an algorithmic context, though the reasons for doing so vary. The most prominent reason in this dissertation for studying randomized algorithms is to broaden the understanding of the connection between conditioning of a problem instance and the performance of iterative algorithms on that problem instance. For example, return to the problem of solving a positive-definite linear system,  $Ax = b$ , equivalently formulated as the optimization problem of minimizing the convex quadratic function  $f(x) = \frac{1}{2}x^T Ax - b^T x$ .

As previously mentioned, the well-studied steepest descent method and conjugate gradient method are both linearly convergent with rates expressible in terms of the relative condition number,  $\|A\| \|A^{-1}\|$ .

For many other iterative algorithms for solving this problem, there is a natural “choice” of search directions to be made at each iteration. From the context of solving the optimization problem, for example, the classical coordinate descent method repeatedly cycles through the set of coordinate directions,  $\{e_1, \dots, e_n\}$ , performing an exact minimization over one variable at each iteration. Alternatively, taking the view of solving the underlying linear system, an alternating projections algorithm cycles through the set of equations and obtain the new iterate by orthogonally projecting the current iterate onto the hyperplane associated with one of the linear equations. These algorithms are of interest because of their low computational cost, each requiring only  $O(n)$  arithmetic operations per iteration. Further, each of these algorithms is known to be linearly convergent, but the rates of convergence are not easily expressible in terms of typical matrix quantities like the condition number. By choosing an appropriate probability distribution over the choice of search directions, however, we can show that the randomized variants of the algorithms satisfy a probabilistic version of linear convergence but with a rate now expressible in terms of classical conditioning concepts.

Naturally, we would like generalizations of the above convergence theory to larger classes of problems. A starting point for such a generalization is to consider arbitrary linear systems  $Ax = b$ , for which we later show that a randomized coordinate descent algorithm is still linearly convergent with a rate dependent on the condition number of  $A$ . A recent result of Strohmer and Vershynin in



[109], slightly extended in Corollary 4.3.9, shows a similar convergence result for a randomized projections algorithm which, interestingly enough, has the same convergence rate as the randomized coordinate descent method. Though this connection may seem surprising, it follows naturally from the fact that the analysis of each algorithm relies on the same error bound for the problem instance.

In fact, iterated projection algorithms have been well-studied for broad classes of problems. Using a similar randomization scheme as for linear equations, we proceed to demonstrate linear convergence for a randomized projection algorithm for linear inequality systems,  $Ax \leq b$ , with a rate expressible in terms of a natural error bound provided by Hoffman in [58]. Further connecting error bounds and distances to ill-posedness, we also provide a distinct convergence rate in terms of the distance of infeasibility to [96]—originally investigated by Renegar and shown to govern the convergence rate of interior point methods—for a specific implementation of this algorithm.

Building upon these results, we continue by considering randomized projection algorithms for convex feasibility problems. After showing that finding a point in the intersection of closed and convex sets can be reformulated into the problem of finding a zero of a specific set-valued mapping, we proceed by using the error bound provided by metric regularity (or metric subregularity) to demonstrate linear convergence for several types of projection-based algorithms.

Observing that the projection operator is actually a special case of the proximal point method leads us in the direction of an even broader problem—finding a zero, or a common zero, of one or more maximal monotone mappings. With regards to the problem of finding a zero of a single mapping, we show that

when the mapping is in fact metrically subregular, the error bound provided by subregularity directly governs the convergence rate of the proximal point algorithm. Even further, similar behavior is shown for the problem of finding a common zero of finitely many maximal monotone operators, via a randomized proximal point algorithm, and a convergence rate is shown that depends both on the error bound derived from the metric subregularity of the mappings themselves as well as from the metric subregularity of the mapping associated with the solution set, similar in nature to what we observe from the randomized projection algorithm for convex feasibility problems. In fact, as a special case of this, we re-obtain the results on randomized projection algorithms.

Although using randomization techniques to demonstrate a broader connection between error bounds, distance to ill-posedness and algorithmic performance is a recurring theme in this dissertation, it's certainly not the only reason for considering randomized algorithms. One practical reason is the hope that certain randomization schemes, even unnatural ones, will lead to improved numerical performance when compared with “traditional” algorithms. In fact, in Chapter 4, we provide examples where seemingly unnatural randomization schemes demonstrate the potential for improved performance on certain classes of problems when compared with either the traditional, deterministic algorithm or the “natural” randomization scheme.

Although Chapters 4 and 5 primarily revolve around the use of randomization to understand how conditioning behavior governs the convergence rate of simple algorithms, the main part of this dissertation begins with an alternate approach. In Chapter 3, we examine a method for estimating the conditioning of a twice-differentiable function in terms of the underlying Hessian matrix. In

particular, through an appropriate randomization scheme, we demonstrate an estimation technique that is linearly convergent (in expectation) to the true Hessian matrix but, unlike many traditional Newton-like methods, does not require direct gradient information. As an application of this technique, we show how a random search algorithm can be accelerated to provide an asymptotic convergence rate independent of the problem’s conditioning. Further, we demonstrate how coordinate descent-style algorithms can be improved to take advantage of the function’s underlying conditioning, leading to superlinear convergence. The “derivative-free” nature of this analysis provides a way of comparing these randomized algorithms with traditional, gradient-based algorithms like steepest descent and Newton-like methods.

Our initial interest in randomized algorithms stems from the seemingly unrelated papers [39] and [42]. In the former paper, certain randomized search schemes are presented as having characteristics of approximation via smoothing, even for discontinuous or non-differentiable problems, while connections with more traditional methods of convex analysis are provided. In the latter paper, a random search technique is used to provide provable error results for differentiable, online minimization problems by using the fact that, in expectation, a certain derivative-free randomization scheme has gradient-like properties. In a sense, the ideas of the latter paper provided the motivation for the results in Chapter 3 while the philosophy behind the former paper—as well as recent work in [109]—encouraged the ideas behind Chapters 4 and 5.

In order to develop the ideas discussed in this introduction more fully, this dissertation is organized as follows. Chapter 2 consists of the common notation, definitions and background material that will be frequently referenced in

the remaining chapters. In Chapter 3, we introduce randomized algorithms for solving twice-differentiable optimization problems and compare the results with traditional methods on a cost-per-function-evaluation basis. In Chapter 4, we consider randomized algorithms for specific classes of problems—positive-definite linear systems, linear equality and inequality systems, and convex feasibility problems—and show how randomization allows the determination of convergence rates in terms of traditional conditioning measures. In Chapter 5, we further examine the interplay between randomization and metric (sub)regularity, developing new convergence theory for proximal point methods.

In conclusion, we would like to say that Chapter 3 is based on a joint paper with A.S. Lewis accepted for publication in the journal *Optimization* at the time of the writing of this dissertation. Chapter 4 is based on a paper with A.S. Lewis submitted for publication to *Mathematics of Operations Research*. Finally, Chapter 5 is based on a paper that has passed through an initial review for the *Journal of Mathematical Analysis and Applications* and is undergoing minor revisions.

## CHAPTER 2

### COMMON NOTATION AND DEFINITIONS

#### 2.1 Introduction

In this chapter, common notation used throughout this dissertation will be defined along with a collection of background results from a variety of mathematical areas. For further reading on some recurring topics, sample references include linear algebra ([59], [60]), convex and variational analysis ([25], [103], [55], [18]), approximation theory ([30]) and probability ([17]).

Throughout this dissertation, unless otherwise stated, assume that  $\mathbb{E}$  is a Hilbert space with an inner product  $\langle \cdot, \cdot \rangle$  and induced norm  $\| \cdot \| = \langle \cdot, \cdot \rangle^{\frac{1}{2}}$ . As frequently used examples of Hilbert spaces, denote the spaces of real numbers,  $n$ -dimensional real vectors and real-valued symmetric  $n \times n$  matrices by  $\mathbb{R}$ ,  $\mathbb{R}^n$ , and  $\mathbb{S}^n$  respectively, each with their usual Euclidean inner products. Also referenced will be the set of non-negative real numbers,  $\mathbb{R}_+$ , and the extended real numbers,  $\bar{\mathbb{R}}$ , defined as  $\mathbb{R} \cup \{\pm\infty\}$ . When necessary, let  $\mathbb{Y}$  be a second Hilbert space whose inner product and norm are denoted identically as above. Whenever possible, we will denote vectors by lowercase letters, constants by lowercase Greek letters, matrices, sets and operators by uppercase letters, random variables by bold text and spaces by “blackboard bold,” like  $\mathbb{E}$ . In the context of randomized algorithms, however, the notation for the underlying probabilistic nature of the iterates will be suppressed for simplicity.

On the Hilbert space  $\mathbb{E}$ , denote the closed unit ball by  $\mathcal{B} = \{x \in \mathbb{E} : \|x\| \leq 1\}$  and the unit sphere by  $\mathbf{S} = \{x \in \mathbb{E} : \|x\| = 1\}$ .

Given two sets  $U$  and  $V$  and  $\alpha \in \mathbb{R}$ , define the set operations element-wise by

$$U + V = \{u + v : u \in U, v \in V\}$$

and

$$\alpha U = \{\alpha u : u \in U\}.$$

## 2.2 Linear Algebra

In what follows, consider  $m$ -by- $n$  real matrices  $A$ . The set of rows of  $A$  is denoted by  $\{a_1^T, \dots, a_m^T\}$  and the set of columns is denoted  $\{A_1, \dots, A_n\}$ . The **spectral norm** of  $A$  is the quantity  $\|A\|_2 := \max_{\|x\|=1} \|Ax\|$  and the **Frobenius norm** is  $\|A\|_F := \sqrt{\sum_{i,j} a_{ij}^2}$ . Additionally, these norms satisfy

$$\|A\|_2 \leq \|A\|_F \leq \sqrt{n}\|A\|_2. \quad (2.2.1)$$

For an arbitrary matrix,  $A$ , let  $\|A^{-1}\|_2$  be the smallest constant  $M$  such that  $\|Ax\|_2 \geq \frac{1}{M}\|x\|_2$  for all vectors  $x$ . In the case  $m \geq n$ , if  $A$  has singular values  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$ , then  $M$  can also be expressed as the reciprocal of the minimum singular value  $\sigma_n$ , and, if  $A$  is invertible, this quantity equals the spectral norm of  $A^{-1}$ . Additionally, denote the smallest non-zero singular value of  $A$  by  $\underline{\sigma}(A)$ .

Now suppose the matrix  $A$  is  $n$ -by- $n$  and positive definite, being symmetric and satisfying  $x^T A x > 0$  for all  $x \neq 0$ . The **energy norm** (or  $A$ -norm), denoted  $\|\cdot\|_A$ , is defined by  $\|x\|_A := \sqrt{x^T A x}$ . This norm satisfies

$$\|x\|_A^2 \leq \|A^{-1}\|_2 \cdot \|Ax\|^2 \text{ for all } x \in \mathbb{R}^n, \quad (2.2.2)$$

$$\|Ax\|^2 \leq \lambda_{\max}(A)\|x\|_A^2 \leq \lambda_{\max}(A)^2\|x\|_2^2 \quad (2.2.3)$$

and

$$\lambda_{\min}(A)\|x\|_2^2 \leq \|x\|_A^2, \quad (2.2.4)$$

where  $\lambda_{\max}(A)$  and  $\lambda_{\min}(A)$  are the maximum and minimum eigenvalues of  $A$ , respectively. Further, if  $A$  is simply positive semi-definite, we can generalize Inequality 2.2.2:

$$x^T Ax \leq \frac{1}{\underline{\lambda}(A)} \|Ax\|^2 \quad (2.2.5)$$

where  $\underline{\lambda}(A)$  is the smallest non-zero eigenvalue of  $A$ . We denote the trace of  $A$  by  $\text{tr } A$ : it satisfies the inequality

$$\|A\|_F \geq \frac{\text{tr } A}{\sqrt{n}}. \quad (2.2.6)$$

For the frequently associated, strictly convex quadratic function  $f(x) = \frac{1}{2}x^T Ax + b^T x$  with minimizer  $x^* = -A^{-1}b$ , the energy norm satisfies

$$\frac{1}{2}\|x - x^*\|_A^2 = f(x) - f(x^*). \quad (2.2.7)$$

Observe that Equation 2.2.7 holds for any solution  $x^*$  to  $Ax = b$  in the case where  $A$  is only positive semi-definite as long as a solution exists, noting that any such solution is a minimizer of  $f$ . However, the left-hand side, defined exactly as before, is no longer technically a norm.

In  $\mathbb{R}^n$ , let  $e_i$  denote the column vector with a 1 in the  $i^{\text{th}}$  position and zeros elsewhere. Additionally, for a vector  $x \in \mathbb{R}^n$ , define the vector  $x^+$  by  $(x^+)_i = \max\{x_i, 0\}$  and the matrix  $\text{Diag}(x)$  to be the matrix whose main diagonal is the vector  $x$  and whose other entries are 0.

Certain conditioning measures for linear systems will be frequently referenced. The **relative condition number** of  $A$  is  $k(A) := \|A\|_2 \|A^{-1}\|_2$ , the commonly used

condition measure. Another measure of interest is the **scaled condition number**, introduced by Demmel in [29], given by  $\kappa(A) := \|A\|_F \|A^{-1}\|_2$ . In particular, these measures are related by

$$k(A) \leq \kappa(A) \leq \sqrt{n} k(A).$$

## 2.3 Convex and Variational Analysis

### 2.3.1 The Basics

Let  $F$  be a **set-valued mapping**, denoted  $F : \mathbb{E} \rightrightarrows \mathbb{Y}$ , such that  $F(x) \subseteq \mathbb{Y}$  for all  $x \in \mathbb{E}$ . The inverse mapping, denoted  $F^{-1}$ , is defined by  $x \in F^{-1}(y) \Leftrightarrow y \in F(x)$ . The **graph**, **domain** and **range** of  $F$ , denoted  $\text{gph } F$ ,  $\text{dom } F$  and  $\text{rng } F$  are defined by

$$\text{gph } F = \{(x, y) : y \in F(x)\},$$

$$\text{dom } F = \{x : F(x) \neq \emptyset\}$$

and

$$\text{rng } F = \cup_{x \in \mathbb{E}} F(x).$$

A set-valued mapping  $F$  is called **single-valued** on a set,  $D \subset \mathbb{E}$ , denoted  $F : D \rightarrow \mathbb{Y}$ , if  $F(x)$  is a singleton for all  $x \in D$ . In such a case, denote  $F(x)$  to be either the single-element set or the unique element of that singleton set as appropriate from the context.

Given a set  $S \subseteq \mathbb{E}$ , the **distance from  $x$  to  $S$** , denoted  $d(x, S)$ , is defined by  $\inf\{\|x - z\| : z \in S\}$ . If  $S$  is closed and convex, define  $P_S(x)$  to be the **projection operator** on  $S$ : that is,  $P_S(x)$  is the unique vector in  $S$  satisfying  $\|x - P_S(x)\| = d(x, S)$ .



**Definition 2.3.1** A single-valued mapping  $T : \mathbb{E} \rightarrow \mathbb{Y}$  is **firmly non-expansive** if

$$\|T(x) - T(y)\|^2 + \|(I - T)(x) - (I - T)(y)\|^2 \leq \|x - y\|^2 \quad \forall x, y \in \mathbb{E} \quad (2.3.2)$$

and **non-expansive** if

$$\|T(x) - T(y)\| \leq \|x - y\| \quad \forall x, y \in \mathbb{E}, \quad (2.3.3)$$

where  $I$  is the identity mapping.

**Proposition 2.3.4** [47, Thm. 12.1] A mapping  $T$  is firmly non-expansive if and only if  $2T - I$  is non-expansive.

**Proposition 2.3.5** The composition of finitely many non-expansive mappings is a non-expansive mapping.

**Proof** Let  $T$  and  $U$  be two non-expansive mappings. Then

$$\|T(U(x)) - T(U(y))\| \leq \|U(x) - U(y)\| \leq \|x - y\|.$$

The result then follows by induction. □

**Proposition 2.3.6** [30, Thm. 5.5] For a closed, convex set,  $S$ , the projection operator  $P_S(\cdot)$  is firmly non-expansive.

By observing that  $P_S(x) = x$  for all  $x \in S$ , the following inequality derived from Inequality 2.3.2 will prove useful later:

$$\|y - x\|^2 - \|P_S(y) - x\|^2 \geq \|y - P_S(y)\|^2 \quad \text{for all } x \in S, y \in \mathbb{E}. \quad (2.3.7)$$

A central tool in convex analysis is that of the normal cone.

**Definition 2.3.8** The *normal cone* to a closed, convex set  $S$  at  $x$  is defined as  $N_S(x) = \emptyset$  if  $x \notin S$  and, if  $x \in S$ ,

$$N_S(x) := \{y \in \mathbb{E} : \langle y, s - x \rangle \leq 0 \ \forall s \in S\}. \quad (2.3.9)$$

The projection operator can be characterized in terms of the normal cone.

**Proposition 2.3.10** [30, Thm. 4.1] For a closed convex set,  $S$ , the corresponding projection operator can be characterized by

$$z = P_S(x) \text{ if and only if } z \in S \text{ and } x - z \in N_S(z). \quad (2.3.11)$$

Specifically,  $x - P_S(x) \in N(P_S(x))$  for all  $x$ .

**Definition 2.3.12** Given a single-valued function  $f : \mathbb{E} \rightarrow \bar{\mathbb{R}}$ , the *epigraph* of  $f$  is defined to be

$$\text{epi} f = \{(x, y) \in \mathbb{E} \times \mathbb{R} : y \geq f(x)\}.$$

**Definition 2.3.13** A single-valued function  $f : \mathbb{E} \rightarrow \bar{\mathbb{R}}$  is *convex* if its epigraph is a convex set.

Additionally, for a single-valued function  $f$ , the domain of  $f : \mathbb{E} \rightarrow \bar{\mathbb{R}}$  is defined to be the domain of the mapping whose graph is the epigraph of  $f$ .

**Definition 2.3.14** The *subdifferential* of a convex function  $f$  at  $\bar{x}$ , denoted  $\partial f(\bar{x})$ , is defined by

$$\partial f(\bar{x}) = \{y \in \mathbb{E} : f(x) \geq f(\bar{x}) + \langle y, x - \bar{x} \rangle \text{ for all } x \in \mathbb{E}\}$$

for  $\bar{x} \in \text{dom} f$  and  $\partial f(\bar{x}) = \emptyset$  otherwise.

**Example 2.3.15** Let  $\iota_S$  be the *indicator function* for  $S \subseteq \mathbb{E}$ , where  $S$  is a closed and convex set, satisfying  $\iota_S(x) = 0$  when  $x \in S$  and  $\iota_S(x) = \infty$  otherwise. Then  $\partial \iota_S(\bar{x}) = N_S(\bar{x})$ .

Let  $\mathbf{X}$  be a random vector on  $\mathbb{E}$  and let  $\mathbf{E}[\cdot]$  be the expected value operator with respect to the probability distribution of  $\mathbf{X}$ . The following proposition, known as Jensen's Inequality, will be frequently used.

**Proposition 2.3.16 (Jensen's Inequality)** Let  $X$  be a random vector on  $\mathbb{E}$  and let  $f : \mathbb{E} \rightarrow \bar{\mathbb{R}}$  be a convex function. Then

$$f(\mathbf{E}[X]) \leq \mathbf{E}[f(X)].$$

### 2.3.2 Metric Regularity and Subregularity

Consider a set-valued mapping  $\Phi : \mathbb{E} \rightrightarrows \mathbb{Y}$  and the problem of solving the associated constraint system  $\bar{b} \in \Phi(x)$  for the unknown vector  $x$ . Building upon the idea of an error bound for linear systems as discussed in Chapter 1, we consider related regularity conditions for set-valued mappings. The first is that of metric regularity.

**Definition 2.3.17** The set-valued mapping  $\Phi : \mathbb{E} \rightrightarrows \mathbb{Y}$  is *metrically regular* at  $\bar{x}$  for  $\bar{b}$  if  $\bar{b} \in \Phi(\bar{x})$  and there exists  $\gamma > 0$  such that

$$d(x, \Phi^{-1}(b)) \leq \gamma d(b, \Phi(x)) \quad \text{for all } (x, b) \text{ near } (\bar{x}, \bar{b}). \quad (2.3.18)$$

The *modulus of regularity*, denoted  $\text{Reg } \Phi(\bar{x}|\bar{b})$ , is the infimum of all constants  $\gamma$  such that Inequality 2.3.18 holds.

Metric regularity generalizes the error bounds previously discussed at the expense of only guaranteeing a bound in local terms. For example, if  $\Phi$  is a single-valued linear map, then the modulus of regularity (at any  $\bar{x}$  for any  $\bar{b}$ ) corresponds to the typical conditioning measure  $\|\Phi^{-1}\|_2$  (with  $\|\Phi^{-1}\|_2 = \infty$  implying the map is not metrically regular) and if  $\Phi$  is a smooth single-valued mapping, then the modulus of regularity is the reciprocal of the minimum singular value of the Jacobian,  $\nabla\Phi(x)$ .

The property of metric regularity possesses strong connections with other ideas in variational analysis. The simplest is that it provides a generalization of the Banach open mapping principle which effectively says, as shown in [32, Ex. 1.1], that a bounded and linear mapping is metrically regular if and only if it is surjective, in which case the modulus of regularity is simply  $\sup\{d(0, A^{-1}(y)) : y \in \mathcal{B}\}$ . If the mapping  $\Phi$  has a closed-convex graph, the Robinson-Ursescu Theorem ([111], [100], et. al.) says that  $\Phi$  is metrically regular at  $\bar{x}$  for  $\bar{b}$  if and only if  $\bar{b}$  is in the interior of the range of  $\Phi$ . Metric regularity is additionally known to be equivalent to several other properties in variational analysis, namely the Aubin property of  $\Phi^{-1}$  and the openness at linear rate of  $\Phi$  ([103, Thm 9.43]). Further, a result originating with Lyusternik and Graves ([79], [49]) and extended by others (for example, [31], [61], [32]) shows that metric regularity is determined by the first-order behavior of a mapping and is preserved under perturbations of mappings with sufficiently small Lipschitz constant. Additional information about metric regularity and its relationship to other concepts in variational analysis can be found in the surveys [33], [61], among others.

From an alternative perspective, metric regularity provides a framework for generalizing the Eckart-Young result on the distance to ill-posedness of linear

mappings cited in Theorem 1.0.1.

**Definition 2.3.19** For a set-valued mapping  $\Phi : \mathbb{E} \rightrightarrows \mathbb{Y}$  with closed graph, the **radius of metric regularity** at  $\bar{x}$  for  $\bar{b}$  is given by

$$\text{rad } \Phi(\bar{x}|\bar{b}) = \inf\{\|G\| : \Phi + G \text{ not metrically regular at } \bar{x} \text{ for } \bar{b} + G(\bar{x})\},$$

where the infimum is over all linear mappings  $G$ .

The following strikingly simple relationship between the radius of regularity and the modulus of regularity was shown in [32].

**Proposition 2.3.20**

$$\text{rad } \Phi(\bar{x}|\bar{b}) \geq \frac{1}{\text{Reg } \Phi(\bar{x}|\bar{b})},$$

with equality holding when  $\Phi$  is a mapping between finite dimensional spaces.

A slightly weaker condition than metric regularity is that of metric subregularity, defined as in [62].

**Definition 2.3.21** The set-valued mapping  $\Phi : \mathbb{E} \rightrightarrows \mathbb{Y}$  is **metrically subregular** at  $\bar{x}$  for  $\bar{b} \in \Phi(\bar{x})$  if there exists  $\gamma > 0$  such that

$$d(x, \Phi^{-1}(\bar{b})) \leq \gamma d(\bar{b}, \Phi(x)) \text{ for all } x \text{ near } \bar{x}. \quad (2.3.22)$$

The **modulus of subregularity**, denoted  $\text{Subreg } \Phi(\bar{x}|\bar{b})$ , is the infimum of all constants  $\gamma$  such that Inequality 2.3.22 holds.

Observe that the reference vector  $\bar{b}$  is fixed in Inequality 2.3.22 for metric subregularity, but not in Inequality 2.3.18 for metric regularity; from this, it naturally follows that  $\text{Subreg } \Phi(\bar{x}|\bar{b}) \leq \text{Reg } \Phi(\bar{x}|\bar{b})$ . In [33], the following, slightly modified definition of metric subregularity is used instead.

**Definition 2.3.23 ([33])** *The set-valued mapping  $\Phi : \mathbb{E} \rightrightarrows \mathbb{Y}$  is **metrically subregular** at  $\bar{x}$  for  $\bar{b} \in \Phi(\bar{x})$  if there exists  $\gamma > 0$  and a neighborhood  $V$  of  $\bar{b}$  such that*

$$d(x, \Phi^{-1}(\bar{b})) \leq \gamma d(\bar{b}, \Phi(x) \cap V) \text{ for all } x \text{ near } \bar{x}. \quad (2.3.24)$$

As noted without proof in [62], the definitions are equivalent in the sense that, given  $\bar{x}$  and  $\bar{b} \in \Phi(\bar{x})$ , Definition 2.3.21 holds if and only if Definition 2.3.23 holds. We include a short proof of this equivalence here for completeness.

**Proposition 2.3.25** *The set-valued mapping  $\Phi : \mathbb{E} \rightrightarrows \mathbb{Y}$  is metrically subregular at  $\bar{x}$  for  $\bar{b}$  according to Definition 2.3.21 if and only if it is metrically subregular at  $\bar{x}$  for  $\bar{b}$  according to Definition 2.3.23.*

**Proof**  $\Rightarrow$ : Suppose there exists  $\gamma > 0$  such that Definition 2.3.21 holds. Then, choosing  $V = \mathbb{Y}$ , the result follows trivially.

$\Leftarrow$ : Let  $x$  be sufficiently near  $\bar{x}$  so that Definition 2.3.23 holds with constant  $\gamma > 0$  and note that, if  $\Phi(x) = \emptyset$ , then Inequality 2.3.22 holds trivially. Therefore, assume  $\Phi(x) \neq \emptyset$ . To temporarily abuse some previous notation, define  $P_{\Phi(x)}(\bar{b})$  to be any element of  $\mathbb{E}$  that attains the infimum of  $\inf\{\|y - \bar{b}\| : y \in \text{cl}(\Phi(x))\}$ , where  $\text{cl}(S)$  is the closure of  $S$ , implying that  $d(\bar{b}, \Phi(x)) = \|\bar{b} - P_{\Phi(x)}(\bar{b})\|$ . From this, it follows that

$$\begin{aligned} d(x, \Phi^{-1}(\bar{b})) &\leq \gamma d(\bar{b}, \Phi(x) \cap V) \quad (\text{Inequality 2.3.24}) \\ &\leq \gamma [\|\bar{b} - P_{\Phi(x)}(\bar{b})\| + d(P_{\Phi(x)}(\bar{b}), V)] \quad (\text{Triangle Inequality}) \\ &\leq 2\gamma \|\bar{b} - P_{\Phi(x)}(\bar{b})\| \quad (\text{since } \bar{b} \in V) \\ &= 2\gamma d(\bar{b}, \Phi(x)) \quad (\text{Definition of Projection}). \end{aligned}$$

□

Metric subregularity of  $\Phi$  is shown to be related to the calmness of  $\Phi^{-1}$  and this relationship is thoroughly explored in several papers, including [62] and [115]. Unfortunately, metric subregularity fails to imply many of the stability properties implied by metric regularity. Examples are shown in [33] where metric subregularity is not preserved under a perturbation with Lipschitz constant 0, unlike metric regularity. Further, examples are given that show that metric subregularity implies no “natural” relationship between the modulus and the radius of subregularity like the one of Proposition 2.3.20.

### 2.3.3 Geometry and Metric Regularity

Given closed and convex sets  $S_1, \dots, S_m \subseteq \mathbb{E}$ , we often want to consider regularity aspects of the sets themselves. We will examine one approach that involves considering regularity properties of a related set-valued mapping. Endow the product space  $\mathbb{E}^m$  with the inner product

$$\langle (u_1, u_2, \dots, u_m), (v_1, v_2, \dots, v_m) \rangle = \sum_{i=1}^m \langle u_i, v_i \rangle$$

and consider the set-valued mapping  $\Phi : \mathbb{E} \rightrightarrows \mathbb{E}^m$  given by

$$\Phi(x) = [S_1 - x, S_2 - x, \dots, S_m - x]^T. \quad (2.3.26)$$

Then it clearly follows that  $\bar{x} \in \cap_i S_i$  if and only if  $0 \in \Phi(\bar{x})$ . Using metric regularity as a starting point, suppose  $\Phi(x)$  is metrically regular at  $\bar{x}$  for 0. From the definition, this is equivalent to the *strong metric inequality*, examined in [67] and [68], among others, defined by the existence of  $\beta, \delta > 0$  such that, for  $i = 1, \dots, m$ ,

$$d(x, \cap_i (S_i - z_i)) \leq \beta \max_{1 \leq i \leq m} d(x + z_i, S_i) \text{ for all } x \in \bar{x} + \delta \mathcal{B}, z_i \in \delta \mathcal{B}. \quad (2.3.27)$$

Characterizing this in terms of normal cones, it was shown in [68, Thm. 1, Prop. 10, Cor. 2] that this is equivalent to the existence of constants  $\delta, k > 0$  such that

$$z_i \in \delta\mathcal{B}, y_i \in N_{S_i}(\bar{x} + z_i) \ (i = 1, \dots, m) \Rightarrow \sum_i \|y_i\|^2 \leq k^2 \left\| \sum_i y_i \right\|^2. \quad (2.3.28)$$

By using the formula in [103, Thm 9.43] for expressing the modulus of regularity in terms of coderivatives, it was shown in [70] that

$$\text{Reg}\Phi(\bar{x}|0) = \lim_{\delta \downarrow 0} \left\{ \inf\{k : \text{Inequality 2.3.28 holds.}\} \right\}. \quad (2.3.29)$$

As a corollary to Equation 2.3.29, we obtain the following result, which will be useful later, that nicely rephrases Equation 2.3.28.

**Corollary 2.3.30 ([70])** *Suppose the set-valued mapping  $\Phi(x) = [S_1 - x, \dots, S_m - x]^T$  is metrically regular at  $\bar{x}$  for 0 and let  $\bar{\gamma}$  be any constant greater than  $\text{Reg}\Phi(\bar{x}|0)$ . Then for all  $x_i \in S_i$  sufficiently near  $\bar{x}$ , any vectors  $y_i \in N_{S_i}(x_i)$ ,  $i = 1, \dots, m$  satisfy*

$$\sum_i \|y_i\|^2 \leq \bar{\gamma}^2 \left\| \sum_i y_i \right\|^2.$$

Consider a relaxed variant of the strong metric inequality, known simply as the *metric inequality* as studied in [61], [86] and [68] among others, defined to hold at  $\bar{x}$  if there exists  $\beta > 0$  such that

$$d(x, \cap_i S_i) \leq \beta \max_{1 \leq i \leq m} d(x, S_i) \text{ for all } x \in \bar{x} + \delta\mathcal{B}. \quad (2.3.31)$$

If Inequality 2.3.31 is valid for  $\delta = \infty$ , we obtain the property of linear regularity and if it holds for all  $\delta > 0$ , it is equivalent to the property of bounded linear regularity, as studied in [7], [8], [9], [10], [15] and others, often in an algorithmic context. In the following result, we see that the existence of a  $\delta > 0$  such that Inequality 2.3.31 holds is equivalent to the previously defined mapping  $\Phi$  being metrically subregular at  $\bar{x}$  for 0.



**Proposition 2.3.32** *Given a collection of closed, convex sets  $\{S_1, \dots, S_m\}$ , the set-valued function  $\Phi(x) = [S_1 - x, \dots, S_m - x]^T$  is metrically subregular at  $\bar{x}$  for 0 if and only if there exist  $\beta, \delta > 0$  such that Inequality 2.3.31 holds.*

**Proof**  $\Rightarrow$ : Suppose  $\Phi$  is metrically subregular at  $\bar{x}$  for 0 with constant  $\kappa$ . Then there exists a neighborhood of  $\bar{x}$  such that:

$$\begin{aligned} d(x, \cap_i S_i)^2 &= d(x, \Phi^{-1}(0))^2 \leq \kappa^2 d(0, \Phi(x))^2 \\ &= \kappa^2 \sum_i d(x, S_i)^2 \leq m\kappa^2 \max_i \{d(x, S_i)^2\}. \end{aligned}$$

Hence, there exists a neighborhood of  $\bar{x}$  such that Inequality 2.3.31 holds.

$\Leftarrow$ : Suppose there exists  $\delta > 0$  such that Inequality 2.3.31 holds with constant  $\beta$ .

Then, for all  $x \in \bar{x} + \delta\mathcal{B}$ ,

$$\begin{aligned} d(x, \Phi^{-1}(0))^2 &= d(x, \cap_i S_i)^2 \leq \beta^2 \max_i \{d(x, S_i)^2\} \\ &\leq \beta^2 \sum_i d(x, S_i)^2 = \beta^2 d(0, \Phi(x))^2, \end{aligned}$$

implying metric subregularity of  $\Phi$ . □

## 2.4 Linear Convergence

In this section, definitions regarding the convergence of sequences will be provided. In what follows, assume  $S \subseteq \mathbb{E}$  is a convex set and let  $\rho : \mathbb{E} \rightarrow \mathbb{R}_+$  be an arbitrary norm on  $\mathbb{E}$  (i.e., any function satisfying  $\rho(x) = 0$  if and only if  $x = 0$ ,  $\rho(\lambda x) = |\lambda|\rho(x)$  and  $\rho(x + y) \leq \rho(x) + \rho(y)$  for all  $\lambda \in \mathbb{R}$ ,  $x, y \in \mathbb{E}$ ). Further, for  $x \in \mathbb{E}$ , define the  **$\rho$ -distance from  $x$  to  $S$**  by  $d_\rho(x, S) = \inf\{\rho(x - y) : y \in S\}$ .

For notational simplicity, if no norm  $\rho$  is specified, take  $\rho$  to be the norm induced by the inner product on  $\mathbb{E}$  (e.g.  $\rho(x) = \langle x, x \rangle^{\frac{1}{2}}$ ), in which case, the definition of  $d_\rho(x, S)$  matches the one given in Section 2.3. Using these concepts, we can proceed to define various methods of convergence.

**Definition 2.4.1** Let  $\{x_j\}_{j \geq 0} \subseteq \mathbb{E}$  be a sequence of vectors and  $\rho$  a norm on  $\mathbb{E}$ . Then  $\{x_j\}$  is **linearly convergent to  $S$  with respect to  $\rho$**  if there exists a constant  $\alpha \in [0, 1)$  such that, for all  $j \geq 0$ ,

$$d_\rho(x_{j+1}, S) \leq \alpha d_\rho(x_j, S).$$

**Definition 2.4.2** Let  $\{x_j\}_{j \geq 0} \subseteq \mathbb{E}$  be a sequence of vectors and  $\rho$  a norm on  $\mathbb{E}$ . Then  $\{x_j\}$  is **super-linearly convergent to  $S$  with respect to  $\rho$**  if either  $x_j \in S$  for all  $j$  sufficiently large, or

$$\lim_{j \rightarrow \infty} \frac{d_\rho(x_{j+1}, S)}{d_\rho(x_j, S)} = 0.$$

When discussing a random vector, we will denote the expected value with respect to the underlying probability distribution by  $\mathbf{E}[\cdot]$ . In this case, we obtain the following generalized definition of linear convergence.

**Definition 2.4.3** Let  $\{X_j\}_{j \geq 0}$  be a sequence of random vectors and  $\rho$  a norm on  $\mathbb{E}$ . Then  $\{X_j\}$  is **linearly convergent in expectation to  $S$  with respect to  $\rho$**  if there exists a constant  $\alpha \in [0, 1)$  such that, for all  $j \geq 0$ ,

$$\begin{aligned} d_\rho(X_{j+1}, S) &\leq d_\rho(X_j, S) \text{ with probability } 1 \\ \mathbf{E}[d_\rho(X_{j+1}, S)^2 \mid X_j] &\leq \alpha d_\rho(X_j, S)^2. \end{aligned}$$

A more commonly used notion of convergence of random variables is that of almost sure convergence, defined as follows.

**Definition 2.4.4** A sequence of random vectors  $\{\mathbf{X}_j\}_{j \geq 0}$  converges *almost surely* to the random vector  $\mathbf{X}$  if

$$P(\lim_{j \rightarrow \infty} \mathbf{X}_j = \mathbf{X}) = 1.$$

The next result provides an initial characterization of the probabilistic consequences of linear convergence in expectation.

**Proposition 2.4.5** Suppose the sequence of random vectors,  $\{\mathbf{X}_j\}_{j \geq 0}$  is linearly convergent in expectation to  $S$  with respect to  $\rho$  and that the random variable  $d_\rho(\mathbf{X}_0, S)$  is bounded above almost surely. Then  $\lim_{j \rightarrow \infty} d_\rho(\mathbf{X}_j, S) = 0$  almost surely.

**Proof** For  $j = 0, 1, 2, \dots$ , define  $Y_j = d_\rho(\mathbf{X}_j, S)$ . By assumption,  $Y_j$  is non-negative and monotonically non-increasing, implying that  $Y_j$  converges to some non-negative random variable  $Y$  almost surely (see, for example, [17]). Further, we know that

$$\mathbf{E}[Y_{j+1}^2 \mid X_0] = \mathbf{E}[\mathbf{E}[Y_{j+1}^2 \mid X_j] \mid X_0] \leq \mathbf{E}[\alpha Y_j^2 \mid X_0]$$

by assumption for some  $\alpha \in [0, 1)$ . By induction, it follows that

$$\mathbf{E}[Y_j^2 \mid X_0] \leq \alpha^j Y_0.$$

Finally, applying the Dominated Convergence Theorem, it follows that

$$\mathbf{E}[Y^2 \mid X_0] = \mathbf{E}[\lim_j Y_j^2 \mid X_0] = \lim_j \mathbf{E}[Y_j^2 \mid X_0] \leq \lim_j \alpha^j Y_0 = 0.$$

From  $\mathbf{E}[Y^2 \mid X_0] = 0$  and  $Y \geq 0$  almost surely, we can conclude that  $Y = 0$  almost surely. □

## CHAPTER 3

### RANDOMIZED HESSIAN ESTIMATION

#### 3.1 Introduction

Stochastic techniques in directional search algorithms have been well-studied in solving optimization problems, often where the underlying functions themselves are random or noisy. For example, some of these algorithms are based on directional search methods that obtain a random search direction which approximates a gradient in expectation. For some background on this class of algorithms, see [40, Ch. 6] or [108, Ch. 5]. In general, for many randomized algorithms, the broad convergence theory, combined with inherent computational simplicity, makes them particularly appealing, even for noiseless, deterministic optimization problems.

In this chapter, we avoid any direct use of gradient information, relying only on function evaluations. In that respect, the methods we consider have the flavor of derivative-free algorithms. Our goal, however, is not the immediate development of a practical, competitive, derivative-free optimization algorithm: our aim is instead primarily speculative. In contrast with much of the derivative-free literature, we make several impractical assumptions that hold throughout this chapter. We assume that the function we seek to minimize is twice differentiable and that evaluations of that function are reliable, cheap, and accurate. Further, we assume that derivative information is neither available directly nor via automatic differentiation, but it is well-approximated by finite differencing. Additionally, we assume that any line search subproblem is relatively cheap to solve when compared to the cost of approximating a gradient. This last as-

sumption is based on the fact that, asymptotically, the computational cost of a line search should be independent of the problem dimension, being a one-dimensional optimization problem, while the number of function evaluations required to obtain a gradient through finite differencing grows linearly with the problem dimension. Within this narrow framework, we consider the question as to whether, in principle, randomization can be incorporated to help simple iterative algorithms achieve good asymptotic convergence.

Keeping this narrow framework in mind, this chapter is organized as follows. In the remainder of this section, we consider a randomized directional search algorithm that chooses a search direction uniformly at random from the unit sphere and apply it to convex quadratic functions, comparing convergence results with a traditional gradient descent algorithm. In Section 3.2, we introduce a technique of randomized Hessian estimation and prove some basic properties. In Section 3.3, we consider algorithmic applications of our randomized Hessian estimation method. In particular, we show how Hessian estimates can be used to accelerate the uniformly random search algorithm introduced in this section and, additionally, how randomized Hessian estimation can also be used to develop a conjugate direction-like algorithm.

As an initial illustration of the use of randomization, consider the following basic algorithm: at each iteration, choose a search direction uniformly at random on the unit sphere and perform an exact line search. This algorithm itself has been widely studied, with analysis appearing in [45] and [105], among others. Further, it was shown to be linearly convergent for twice differentiable functions under conditions given in [94].

Consider applying this algorithm to the problem of minimizing a convex

quadratic function  $f(x) = \frac{1}{2}x^T Ax + b^T x$  where  $A$  is a positive-definite,  $n \times n$  matrix.

Observe that if the current iterate is  $x$ , then the new iterate is given by

$$x_+ = x - \frac{d^T(Ax + b)}{d^T Ad} d \quad (3.1.1)$$

and the new function value is

$$f(x_+) = f(x) - \frac{(d^T(Ax + b))^2}{2d^T Ad}.$$

The difference between the current function value and the optimal value is reduced by the ratio

$$\begin{aligned} \frac{f(x_+) - f(x^*)}{f(x) - f(x^*)} &= 1 - \frac{(d^T(Ax + b))^2}{2(d^T Ad)(f(x) - f(x^*))} \\ &= 1 - \frac{(d^T(Ax + b))^2}{(d^T Ad)((x - x^*)^T A(x - x^*))} \\ &= 1 - \frac{(d^T A(x - x^*))^2}{(d^T Ad)((A(x - x^*))^T A^{-1}(A(x - x^*)))} \\ &\leq 1 - \frac{1}{k(A)} \left( d^T \frac{A(x - x^*)}{\|A(x - x^*)\|} \right)^2. \end{aligned}$$

Observe that the distribution of  $d$  is invariant under orthogonal transformations. Therefore, let  $U$  be any orthogonal transformation satisfying  $U\left(\frac{A(x - x^*)}{\|A(x - x^*)\|}\right) = e_1$ , the first standard basis vector. From this, we have

$$\begin{aligned} \mathbf{E}\left[\left(d^T \frac{A(x - x^*)}{\|A(x - x^*)\|}\right)^2 \mid x\right] &= \mathbf{E}\left[\left((U^T d)^T \frac{A(x - x^*)}{\|A(x - x^*)\|}\right)^2 \mid x\right] \\ &= \mathbf{E}[d_1^2] \\ &= \frac{1}{n} \mathbf{E}\left[\sum_i d_i^2\right] \\ &= \frac{1}{n}, \end{aligned}$$

where the first equality follows from the invariance of the distribution of  $d$  and the third equality follows from the fact that each component of  $d$  is identically

distributed. We deduce

$$\mathbf{E}[f(x_+) - f(x^*) \mid x] \leq \left(1 - \frac{1}{n k(A)}\right)(f(x) - f(x^*)) \quad (3.1.2)$$

with equality when  $A$  is a multiple of the identity matrix, in which case  $k(A) = 1$ .

Compare this with the steepest descent algorithm. A known result about the steepest descent algorithm in [1] says that, given initial iterate  $x$  and defining  $\hat{x}$  to be the new iterate constructed from an exact line search in the negative gradient direction,

$$f(\hat{x}) - f(x^*) \leq \left(\frac{k(A) - 1}{k(A) + 1}\right)^2 (f(x) - f(x^*)) = \left(1 - O\left(\frac{1}{k(A)}\right)\right)(f(x) - f(x^*)).$$

Further, for most initial iterates  $x$ , this inequality is asymptotically tight if this procedure is iteratively repeated. Consider the following asymptotic argument, applying the assumptions made earlier in this section. Suppose derivative information is only available through—and well-approximated by—finite differencing but we can perform an exact (or almost-exact) line search in some constant number,  $O(1)$ , of function evaluations. It follows that each iteration of random search takes  $O(1)$  function evaluations. However, since derivative information is only available via finite differencing, computing a gradient takes  $O(n)$  function evaluations. Letting  $\bar{x}$  be the iterate after performing  $O(n)$  iterations of random search, we obtain that

$$\mathbf{E}\left[\frac{f(\bar{x}) - f(x^*)}{f(x) - f(x^*)} \mid x\right] \leq \left(1 - \frac{1}{n k(A)}\right)^{O(n)} = 1 - O\left(\frac{1}{k(A)}\right).$$

Essentially, the expected improvement of random search is on the same order of magnitude as steepest descent when measured on a cost per function evaluation basis. This simple example suggests that randomization techniques may

be an interesting ingredient in the design and analysis of iterative optimization algorithms.

### 3.2 Randomized Hessian Estimation

In this section, we will consider arbitrary twice-differentiable functions  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ . As in the previous section, assume these functions can be evaluated exactly, but derivative information is only available through finite differencing. In particular, for any vector  $v \in \mathbb{R}^n$ , suppose we can use finite differencing to well-approximate the second derivative of  $f$  at  $x$  in the direction  $v$  via the formula

$$v^T \nabla^2 f(x) v \approx \frac{f(x + \epsilon v) - 2f(x) + f(x - \epsilon v)}{\epsilon^2} \quad (3.2.1)$$

for some sufficiently small  $\epsilon > 0$ . In particular, note that by choosing  $\frac{1}{2}n(n+1)$  suitable directions  $v$ , we could effectively approximate the entire Hessian  $\nabla^2 f(x)$ .

In Section 3.1, we considered a framework in which computational costs of an algorithm are measured by the number of function evaluations required and we will continue with that throughout this chapter. In particular, it was shown that under this framework, the steepest descent algorithm, asymptotically, achieves improvement on the same order of magnitude as a uniformly random search algorithm when applied to convex quadratics. Ideally, we would like to extend these methods of analysis to algorithms that incorporate additional information about a function's behavior. For example, instead of calculating a complete Hessian matrix at each iteration, Newton-like methods rely on approximations to the Hessian matrix which are iteratively updated, often from successively generated gradient information. To consider a similar approach in the context



of random search, suppose we begin with an approximation to the Hessian matrix, denoted  $B$ , and some unit vector  $v \in \mathbb{R}^n$ . Consider the new matrix  $B_+$  obtained by making a rank-one update so that the new matrix  $B_+$  matches the true Hessian in the direction  $v$ , i.e.,

$$B_+ = B + (v^T(\nabla^2 f(x) - B)v)vv^T. \quad (3.2.2)$$

This rank-one update results in the new matrix  $B_+$  having the property that  $v^T B_+ v = v^T \nabla^2 f(x) v$ . Note that if this update is performed using the approximate second derivative via Equation 3.2.1, then this only costs 3 function evaluations. For the remainder of this section, assume the space of symmetric  $n \times n$  matrices,  $\mathbb{S}^n$ , is equipped with the usual trace inner product  $\langle X, Y \rangle = \text{tr}(X^T Y)$  and the induced Frobenius norm. We proceed with the following result.

**Theorem 3.2.3** *Given any matrices  $H, B \in \mathbb{S}^n$ , if the random vector  $d \in \mathbb{R}^n$  is uniformly distributed on the unit sphere, then the matrix*

$$B_+ = B + (d^T(H - B)d)dd^T$$

*satisfies*

$$\|B_+ - H\| \leq \|B - H\|$$

*and*

$$\mathbf{E}[\|B_+ - H\|^2] \leq \left(1 - \frac{2}{n(n+2)}\right)\|B - H\|^2.$$

**Proof** Since we can rewrite the update in the form

$$(B_+ - H) = (B - H) - (d^T(B - H)d)dd^T,$$

we lose no generality in assuming  $H = 0$ . Additionally, we lose no generality in assuming  $\|B\| = 1$ , and proving

$$\|B_+\| \leq 1 \text{ and } \mathbf{E}[\|B_+\|^2] \leq 1 - \frac{2}{n(n+2)}.$$

From the equation

$$B_+ = B - (d^T B d) d d^T,$$

we immediately deduce

$$\|B_+\|^2 = \|B\|^2 - (d^T B d)^2 = 1 - (d^T B d)^2 \leq 1.$$

To complete the proof, we need to bound the quantity  $\mathbf{E}[(d^T B d)^2]$ . We can diagonalize the matrix  $B = U^T (\text{Diag } \lambda) U$  where  $U$  is orthogonal and the vector of eigenvalues  $\lambda \in \mathbb{R}^n$  satisfies  $\|\lambda\| = 1$  by assumption. Using the fact that the distribution of  $d$  is invariant under orthogonal transformations, we obtain

$$\begin{aligned} \mathbf{E}[(d^T B d)^2] &= \mathbf{E}[(d^T U^T (\text{Diag } \lambda) U d)^2] = \mathbf{E}[(d^T (\text{Diag } \lambda) d)^2] \\ &= \mathbf{E}\left[\left(\sum_{i=1}^n \lambda_i d_i^2\right)^2\right] = \mathbf{E}\left[\sum_i \lambda_i^2 d_i^4 + \sum_{i \neq j} \lambda_i \lambda_j d_i^2 d_j^2\right] \\ &= \mathbf{E}[d_1^4] + \left(\sum_{i \neq j} \lambda_i \lambda_j\right) \mathbf{E}[d_1^2 d_2^2] \end{aligned}$$

by symmetry. Since we know that

$$0 \leq \left(\sum_i \lambda_i\right)^2 = \sum_i \lambda_i^2 + \sum_{i \neq j} \lambda_i \lambda_j = 1 + \sum_{i \neq j} \lambda_i \lambda_j,$$

it follows that

$$\mathbf{E}[(d^T B d)^2] \geq \mathbf{E}[d_1^4] - \mathbf{E}[d_1^2 d_2^2].$$

Standard results on integrals over the unit sphere in  $\mathbb{R}^n$  gives the formula

$$\int_{\|x\|=1} x_1^\nu d\sigma = 2\pi^{\frac{n-1}{2}} \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu+n}{2}\right)},$$

where  $d\sigma$  denotes an  $(n - 1)$ -dimensional surface element, and  $\Gamma(\cdot)$  denotes the Gamma function. We deduce

$$\mathbf{E}[d_1^4] = \frac{\int_{\|x\|=1} x_1^4 d\sigma}{\int_{\|x\|=1} d\sigma} = \frac{\Gamma(\frac{5}{2})}{\Gamma(\frac{n}{2} + 2)} \cdot \frac{\Gamma(\frac{n}{2})}{\Gamma(\frac{1}{2})} = \frac{\frac{3}{2} \cdot \frac{1}{2}}{(\frac{n}{2} + 1) \cdot \frac{n}{2}} = \frac{3}{n(n + 2)}.$$

Furthermore,

$$1 = \left( \sum_i d_i^2 \right)^2 = \sum_i d_i^4 + \sum_{i \neq j} d_i^2 d_j^2,$$

so using symmetry again shows

$$1 = n\mathbf{E}[d_1^4] + n(n - 1)\mathbf{E}[d_1^2 d_2^2].$$

From this we deduce

$$\mathbf{E}[d_1^2 d_2^2] = \frac{1 - n\mathbf{E}[d_1^4]}{n(n - 1)} = \frac{1}{n(n + 2)}.$$

Therefore, this shows that

$$\mathbf{E}[(d^T B d)^2] \geq \frac{3}{n(n + 2)} - \frac{1}{n(n + 2)} = \frac{2}{n(n + 2)},$$

so

$$\mathbf{E}[\|B_+\|^2] \leq 1 - \frac{2}{n(n + 2)}$$

as required. □

To continue, note that iterating this procedure generates a random sequence of Hessian approximations that converges almost surely to the true Hessian, as shown next.

**Corollary 3.2.4** *Given any matrices  $H, B_0 \in \mathbb{S}^n$ , consider the sequence of matrices  $B_k \in \mathbb{S}^n$  for  $k = 0, 1, 2, \dots$ , defined iteratively by*

$$B_{k+1} = B_k + \left( (d^k)^T (H - B_k) d^k \right) d^k (d^k)^T,$$

where the random vectors  $d^0, d^1, d^2, \dots \in \mathbb{R}^n$  are independent and uniformly distributed on the unit sphere. Then the errors  $\|B_k - H\|$  decrease monotonically, and  $B_k \rightarrow H$  almost surely.

**Proof** By Theorem 3.2.3, it follows that the random sequence of matrices  $\{B_k\}$  is linearly convergent in expectation to  $H$  with respect to  $\|\cdot\|_F$ . Therefore, the result follows from Proposition 2.4.5.

In a more realistic framework for optimization, we wish to approximate a limiting Hessian. In the context of randomized algorithms, such as the “random search” algorithm described in Section 3.1, the iterates generated by the algorithm now are random. By using Hessian information at each iterate to update our Hessian estimate, we now have to consider that the corresponding sequence of Hessians used for approximation is now itself random, though ideally approaching a limiting Hessian, in addition to considering the random sequence of Hessian estimates generated by the estimation procedure of Theorem 3.2.3. To account for this in the following theorem, recall that  $\sqrt{\mathbf{E}[\|X\|^2]}$  is a norm on the space of random matrices. Applying properties of norms to this function, as the next result shows, we obtain convergence of the random Hessian estimates to the limiting Hessian.

**Theorem 3.2.5** *Consider a sequence of random matrices  $H_k \in \mathbb{S}^n$  for  $k = 1, 2, 3, \dots$ , with each  $\mathbf{E}[\|H_k\|^2]$  finite, and a fixed matrix  $\bar{H} \in \mathbb{S}^n$  such that  $\mathbf{E}[\|H_k - \bar{H}\|^2] \rightarrow 0$ . Consider a sequence of random matrices  $B_k \in \mathbb{S}^n$  for  $k = 0, 1, 2, \dots$ , with  $\mathbf{E}[\|B_0\|^2]$  finite, related by the iterative formula*

$$B_{k+1} = B_k + \left( (d^k)^T (H_k - B_k) d^k \right) d^k (d^k)^T,$$

where the random vectors  $d^0, d^1, d^2, \dots \in \mathbb{R}^n$  are independent and uniformly distributed on the unit sphere. Then  $\mathbf{E}[\|B_k - \bar{H}\|^2] \rightarrow 0$ .

**Proof** By Corollary 3.2.4, we know for each  $k = 0, 1, 2, \dots$  the inequality

$$\|B_{k+1} - H_k\|^2 \leq \|B_k - H_k\|^2$$

holds. Hence by induction it follows that  $\mathbf{E}[\|B_k\|^2]$  is finite for all  $k \geq 0$ .

Define a number

$$r = \sqrt{1 - \frac{2}{n(n+2)}} \in (0, 1).$$

By Theorem 3.2.3, we have

$$\mathbf{E}[\|B_{k+1} - H_k\|^2 \mid B_k, H_k] \leq r^2 \|B_k - H_k\|^2.$$

Once again, define a probability measure  $\gamma_k$  by

$$\gamma_k(S) = \text{pr}\{(B_k, H_k) \in S\}$$

for any measurable set  $S$ . Then we have

$$\begin{aligned} \mathbf{E}[\|B_{k+1} - H_k\|^2] &= \int \mathbf{E}[\|B_{k+1} - H_k\|^2 \mid (B_k, H_k) = (B, H)] d\gamma_k(B, H) \\ &\leq \int r^2 \|B - H\|^2 d\gamma_k(B, H) \\ &= r^2 \mathbf{E}[\|B_k - H_k\|^2]. \end{aligned}$$

Applying the triangle inequality property of norms gives

$$\sqrt{\mathbf{E}[\|B_{k+1} - \bar{H}\|^2]} \leq r \sqrt{\mathbf{E}[\|B_k - \bar{H}\|^2]} + (1 + r) \sqrt{\mathbf{E}[\|H_k - \bar{H}\|^2]}.$$

Now fix any number  $\epsilon > 0$ . By assumption, there exists an integer  $\bar{k}$  such that for all integers  $k \geq \bar{k}$  we have

$$\mathbf{E}[\|H_k - \bar{H}\|^2] \leq \left( \frac{\epsilon(1-r)}{2(1+r)} \right)^2.$$

Hence, for all  $k \geq \bar{k}$ , we deduce

$$\sqrt{\mathbf{E}[\|B_{k+1} - \tilde{H}\|^2]} \leq r \sqrt{\mathbf{E}[\|B_k - \tilde{H}\|^2]} + \frac{\epsilon(1-r)}{2}.$$

For such  $k$ , if  $\mathbf{E}[\|B_k - \tilde{H}\|^2] \leq \epsilon^2$ , then

$$\sqrt{\mathbf{E}[\|B_{k+1} - \tilde{H}\|^2]} \leq \frac{\epsilon(1+r)}{2} < \epsilon,$$

whereas if  $\mathbf{E}[\|B_k - \tilde{H}\|^2] > \epsilon^2$ , then

$$\begin{aligned} \sqrt{\mathbf{E}[\|B_{k+1} - \tilde{H}\|^2]} &< r \sqrt{\mathbf{E}[\|B_k - \tilde{H}\|^2]} + \frac{1-r}{2} \sqrt{\mathbf{E}[\|B_k - \tilde{H}\|^2]} \\ &= \frac{1+r}{2} \sqrt{\mathbf{E}[\|B_k - \tilde{H}\|^2]}. \end{aligned}$$

Consequently,  $\mathbf{E}[\|B_k - \tilde{H}\|^2] \leq \epsilon^2$  for all large  $k$ . Since  $\epsilon > 0$  was arbitrary, the result follows.  $\square$

### 3.3 Applications to Algorithms

#### 3.3.1 Random Search, Revisited

Return to the convex quadratic function  $f(x) = \frac{1}{2}x^T A x + b^T x$  considered in Section 3.1, where  $A$  is a positive definite,  $n \times n$  matrix and  $x^*$  is the unique minimizer. Recall that if we consider the iterative algorithm given by Equation 3.1.1, letting  $d$  be a unit vector uniformly distributed on the unit sphere and letting

$$x_+ = x - \frac{d^T(Ax + b)}{d^T A d} d,$$

then it was shown in Inequality 3.1.2 that

$$\mathbf{E}[f(x_+) - f(x^*) \mid x] \leq \left(1 - \frac{1}{n k(A)}\right)(f(x) - f(x^*)).$$

Now, suppose that  $H$  is a positive-definite estimate of the matrix  $A$  and consider the Cholesky factor matrix  $C$  such that  $CC^T = H^{-1}$ . Suppose that instead of performing an exact line search in the uniformly distributed direction  $d$ , we instead perform the line search in the direction  $Cd$ . From this we obtain

$$\begin{aligned}
\frac{f(x_+) - f(x^*)}{f(x) - f(x^*)} &= 1 - \frac{(d^T C^T (Ax + b))^2}{2(d^T C^T ACd)(f(x) - f(x^*))} \\
&= 1 - \frac{(d^T C^T (Ax + b))^2}{(d^T C^T ACd)((x - x^*)^T A(x - x^*))} \\
&= 1 - \frac{(d^T (C^T A(x - x^*)))^2}{(d^T (C^T AC)d)((C^T A(x - x^*))^T (C^T AC)^{-1} (C^T A(x - x^*)))} \\
&\leq 1 - \frac{1}{k(C^T AC)} \left( d^T \frac{C^T A(x - x^*)}{\|C^T A(x - x^*)\|} \right)^2,
\end{aligned}$$

allowing us to conclude that

$$\mathbf{E}[f(x_+) - f(x^*) \mid x] \leq \left(1 - \frac{1}{n k(C^T AC)}\right)(f(x) - f(x^*)).$$

This provides the same convergence rate as performing the random search algorithm given by Equation 3.1.1 on the function  $g(x) = \frac{1}{2}x^T (C^T AC)x + b^T x$ .

Consider an implementation of this algorithm using the Hessian approximation technique described in Section 3.2. Given a current iterate  $x_{k-1}$  and Hessian approximation  $B_{k-1}$ , we can proceed as follows. First, form the new Hessian approximation  $B_k$  given by Equation 3.2.2, choosing the update vector uniformly at random from the unit sphere. Observe that by Corollary 3.2.4, if  $A$  is positive definite, then  $B_k$  will be positive definite as well almost surely for all sufficiently large  $k$ , in which case, obtain the Cholesky factorization  $B_k^{-1} = C_k C_k^T$ . Otherwise, one suggested heuristic, implemented below, is to obtain the projection of  $B_k$  onto the positive semi-definite cone, denoted  $B_k^+$ , and perform the

Cholesky factorization  $C_k C_k^T = (B_k^+ + \epsilon I)^{-1}$  for some  $\epsilon > 0$ . Finally, we can find the next iterate  $x_k$  by an exact line search in the direction  $C_k d_k$  where  $d_k$  is uniformly distributed on the unit sphere. Efficient methods for updating the Cholesky factorization can be found in [46].

Since  $B_k \rightarrow A$  almost surely by Corollary 3.2.4, it follows  $C_k \rightarrow A^{-\frac{1}{2}}$  almost surely as well. Therefore, it follows that

$$\frac{\mathbf{E}[f(x_{k+1}) - f(x^*) \mid x_k]}{f(x_k) - f(x^*)} \leq 1 - \frac{1}{n k(C_k^T A C_k)} \rightarrow 1 - \frac{1}{n}.$$

Thus, the uniformly random search algorithm incorporating the Hessian update provides linear convergence with asymptotic rate  $1 - \frac{1}{n}$ , *independent of the conditioning of the original matrix*.

In Figure 3.1, we provide two examples of the algorithm's behavior with a convex quadratic function  $f(x) = \frac{1}{2}x^T A x + b^T x$ , where  $b = [1, 1, \dots, 1]^T$ . The first example uses a Hilbert Matrix of size 7 (with condition number on the order of  $10^8$ ) while the second uses the matrix  $A = \text{Diag}(1, 7, 7^2, \dots, 7^6)$ . In each case, we compare uniformly random search with the Cholesky-weighted random search described above, using the projection heuristic when the Hessian estimate is not positive definite. Additionally, each search vector and Hessian update vector, when applicable, was chosen independently in each example and, when applicable, an exact second derivative calculation was implemented in the Hessian update.

### 3.3.2 A Conjugate Directions Algorithm

Coordinate descent algorithms have a long and varied history in differentiable minimization. In the worst case, examples of continuously differentiable func-



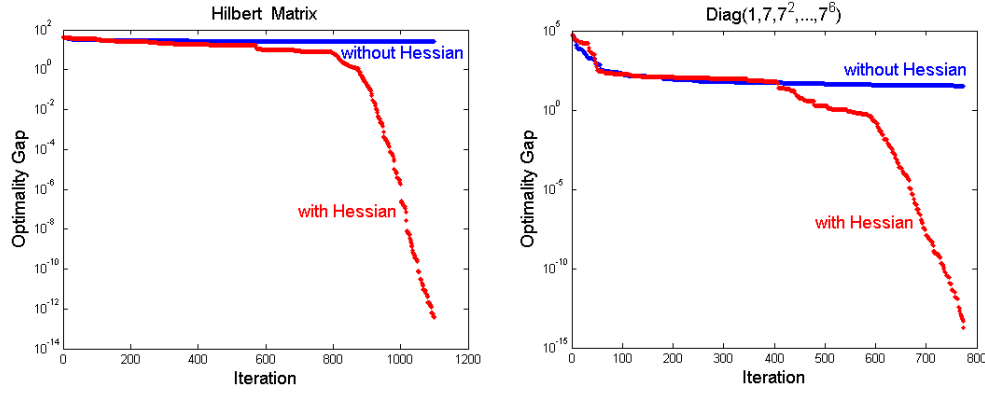


Figure 3.1: Random search with Hessian estimates on convex quadratics

tions exist in [92] where a coordinate descent algorithm will fail to converge to a first-order stationary point. On the other hand, for twice-differentiable, strictly convex functions, variants of coordinate descent methods were shown to be linearly convergent in [76]. In either case, the simplicity of such algorithms, along with the lack of a need for gradient information, often makes them appealing.

Let us briefly return to the example of a convex quadratic function  $f(x) = \frac{1}{2}x^T Ax + b^T x$ . Consider algorithms, similar to coordinate descent algorithms, that choose search directions by cycling through some fixed set  $W = \{w_1, \dots, w_n\}$ , performing an exact line search at each iteration. If the search directions in  $W$  happen to be  $A$ -conjugate, satisfying  $w_i^T A w_j = 0$  for all  $i \neq j$ , then we actually reach the optimal solution in  $n$  iterations. Alternatively, if our set of search directions fails to account for the function's second-order behavior, convergence can be significantly slower. Explicitly generating a set of directions that are conjugate with respect to the Hessian requires knowledge of the function's Hessian information. Methods were proposed in [91], and expanded upon in [113], [19], and [83] among others, that begin as coordinate descent algorithms and iteratively adjust the search directions, gradually making them

conjugate with respect to the Hessian matrix. Further, these adjustments are based on the results of previous line searches without actually requiring full knowledge of the Hessian or any gradients.

We propose an alternative approach for arbitrary twice-differentiable functions. If an estimate of the Hessian were readily available, we could take advantage of it by generating search directions iteratively that are conjugate with respect to the estimate. This suggests that we can design an algorithm using the Hessian estimation technique in Section 3.2 to dynamically generate new search directions that have the desired conjugacy properties. We can formalize this in the following algorithm.

**Algorithm 3.3.1** *Let  $f$  be a twice-differentiable function,  $x_0$  an initial starting point,  $B_0$  an initial Hessian estimate and  $\{v_{-n}, v_{-(n-1)}, \dots, v_{-1}\}$  an initial set of search directions. For  $k = 0, 1, 2, \dots$*

1. *Compute the vector  $v_k$  that is  $B_k$ -conjugate to  $v_{k-1}, \dots, v_{k-n+1}$ .*
2. *Compute  $x_{k+1}$  as a result of a (two-way) line search in the direction  $v_k$ .*
3. *Compute  $B_{k+1}$  according to Equation 3.2.2, letting  $d_k$  be uniformly distributed on the unit sphere and computing*

$$B_{k+1} = B_k + (d_k(\nabla^2 f(x_{k+1}) - B_k)d_k)d_k d_k^T.$$

One simple initialization scheme takes  $B_0 = I$  and  $\{v_{-n}, \dots, v_{-1}\} = \{e_1, \dots, e_n\}$ , the standard basis vectors.

Since  $B_k$  is our Hessian approximation at the current iterate  $x_k$ , two reasonable heuristics for the initial step size are given by  $x_{k+1} = x_k - t_k v_k$ , where  $t_k = \frac{v_k^T \nabla f(x_k)}{v_k^T \nabla^2 f(x_k) v_k}$

or  $t_k = \frac{v_k^T \nabla f(x_k)}{v_k^T B_k v_k}$ , corresponding to an exact line search in the direction  $v_k$  of the exact or estimated quadratic model, respectively. The advantage to this approach is that each iteration requires only directional derivatives and, being highly iterative, this interpolates nicely with the Hessian update derived in Section 3.2. Specifically, when using the fixed step sizes mentioned above, each iteration takes five or four function evaluations, respectively:  $f(x_k)$ ,  $f(x_k \pm \epsilon d_k)$ ,  $f(x_k + \epsilon v_k)$  and, in the first case,  $f(x - \epsilon v_k)$ , where  $v_k$  and  $d_k$  are the search direction and the random unit vector, respectively.

The essence of this algorithm lies in using our randomized Hessian estimation technique to update a quadratic model and then performing a line search. Since we are relying solely on function evaluations, this algorithm has the “flavor” of derivative-free optimization. However, it should be noted that a different perspective can be taken with regards to this algorithm, permitting a comparison with Newton-like methods.

Typical Newton-like methods maintain, along with the current iterate  $x_k$ , a (positive-definite) Hessian estimate  $B_k$  and proceed by performing some type of line search in the direction  $-B_k^{-1} \nabla f(x_k)$ . For simplicity, consider a step size of 1, i.e.,  $x_{k+1} = x_k - B_k^{-1} \nabla f(x_k)$ . Recall that computing  $B_k^{-1} \nabla f(x_k)$ , equivalent to solving the system  $B_k y = \nabla f(x_k)$  for  $y$ , can be done indirectly by searching in  $n$  different  $B_k$ -conjugate directions.

Specifically, suppose we have a set of directions  $\{v_1, \dots, v_n\}$  that are  $B_k$ -conjugate, satisfying  $v_i^T B_k v_j = 0$  for all  $i \neq j$ . and take  $x^0 = x_k$ , our current iterate. For  $i = 1, \dots, n$ , let  $x^i = x^{i-1} - \frac{v_i^T \nabla f(x_k)}{v_i^T B_k v_i} v_i$ . Then it follows that

$$x^n = x^0 - \sum_{i=1}^n \frac{v_i^T \nabla f(x_k)}{v_i^T B_k v_i} v_i = x_k - B_k^{-1} \nabla f(x_k),$$

the Newton-like step. Given this interpretation of Newton-like methods, consider the version of Algorithm 3.3.1 where, at each iteration, the step size is fixed beforehand at  $t_k = \frac{v_k^T \nabla f(x_k)}{v_k^T B_k v_k}$ . Then one can interpret Algorithm 3.3.1 as an iterated version of a Newton-like method. Specifically, while the Newton-like method *indirectly* involves computing the quantities  $v_i^T \nabla f(x_k)$  and  $v_i^T B_k v_i$  with the iterate  $x_k$  and Hessian estimate  $B_k$  fixed, Algorithm 3.3.1 allows for a dynamically changing gradient and Hessian approximation at each conjugate direction step.

Given this connection between Algorithm 3.3.1 and traditional Newton-like methods, it seems natural to expect superlinear convergence under similar assumptions. As we demonstrate in the following result, superlinear convergence is obtained for strictly convex quadratic functions.

**Theorem 3.3.2** *Consider the strictly convex quadratic function  $f(x) = \frac{1}{2}x^T A x + b^T x$  where  $A$  is a positive definite matrix. Then for any initial point  $x_0$ , initial Hessian estimate  $B_0$  and initial search directions, Algorithm 3.3.1 is  $n$ -step superlinearly convergent almost surely when implemented with an exact line search.*

**Proof** Define  $\epsilon_k = \|B_k - A\| = \|B_k - A\|_F$  and note that, by Inequality 2.2.1,  $\|B_k - A\|_2 \leq \epsilon_k$ . Now consider  $n$  consecutive iterations of the algorithm, beginning with iterate  $x_k$  and ending with iterate  $x_{k+n}$ . Without loss of generality, assume the respective search directions satisfy  $\|v_i\| = 1$  for  $i = k, k+1, \dots, k+n-1$ . Recall that by design of the algorithm, these search directions satisfy  $v_i^T B_j v_j = 0$  for any  $j \in \{k, k+1, \dots, k+n-1\}$  and  $i \in \{j-n+1, \dots, j-1\}$ . Note that this implies that for any  $i < j \in \{k, k+1, \dots, k+n-1\}$ ,

$$|v_i^T A v_j| = |v_i^T B_j v_j + v_i^T (A - B_j) v_j| \leq \|v_i\| \|A - B_j\|_2 \|v_j\| \leq \epsilon_k \quad (3.3.3)$$

by Inequality 2.2.1 and the definition of  $\epsilon_k$ .

Next, we construct a matrix  $M_k$  such that these search directions are  $M_k$ -conjugate and  $\|A - M_k\| = O(\epsilon_k)$ . Let  $V_k = [v_k, v_{k+1}, \dots, v_{k+n-1}]$  be the matrix whose columns are the  $n$  consecutive search directions. First, notice that if  $\epsilon_k$  is sufficiently small, this matrix is invertible and the quantity  $\|V_k^{-1}\|_2$  is uniformly bounded. To see this, consider any  $y \in \mathbb{R}^n$  such that  $\|y\| = 1$ . Then,

$$\begin{aligned}
\|V_k y\|_2^2 &= y^T V_k^T V_k y \\
&= (y^T V_k^T A^{\frac{1}{2}}) A^{-1} (A^{\frac{1}{2}} V_k y) \\
&= \|A^{\frac{1}{2}} V^T y\|_{A^{-1}}^2 \\
&\geq \lambda_{\min}(A^{-1}) \|A^{\frac{1}{2}} V_k y\|^2 \quad (\text{by Inequality 2.2.4}) \\
&= \frac{1}{\lambda_{\max}(A)} y^T V_k^T A V_k y \\
&= \frac{1}{\lambda_{\max}(A)} [y^T \text{Diag}(V_k^T A V_k) y + y^T [V_k^T A V_k - \text{Diag}(V_k^T A V_k)] y] \\
&\geq \frac{\lambda_{\min}(A) - n\epsilon_k}{\lambda_{\max}(A)},
\end{aligned}$$

with the last inequality coming from the fact that

$$y^T \text{Diag}(V_k^T A V_k) y = \sum_{i=1}^n y_i^2 v_{k+i-1}^T A v_{k+i-1},$$

Inequality 2.2.4 and Inequality 3.3.3. From the above bound and the alternative definition of  $\|V_k^{-1}\|_2$ , it follows that

$$\|V_k^{-1}\|_2^2 \leq \frac{\lambda_{\max}(A)}{\lambda_{\min}(A) - n\epsilon_k}. \quad (3.3.4)$$

Now consider the matrix  $M_k$  defined by

$$M_k = A - V_k^{-T} (V_k^T A V_k - \text{Diag}(V_k^T A V_k)) V_k^{-1} = V_k^{-T} \text{Diag}(V_k^T A V_k) V_k^{-1}.$$

Further, observe that

$$\begin{aligned}
\|A - M_k\|_2 &= \|V_k^{-T} (V_k^T A V_k - \text{Diag}(V_k^T A V_k)) V_k^{-1}\|_2 \\
&\leq \|V_k^{-1}\|_2^2 \|(V_k^T A V_k - \text{Diag}(V_k^T A V_k))\|_F \\
&\leq \|V_k^{-1}\|_2^2 n\epsilon_k,
\end{aligned}$$

with the first inequality coming from the sub-multiplicity of the spectral norm, the fact that the spectral norm is invariant under matrix transposition and Inequality 2.2.1 while the last inequality comes from Inequality 3.3.3. In particular, the matrix  $M_k$  satisfies  $\|A - M_k\|_2 = O(\epsilon_k)$  and, for  $i \neq j \in \{k, k+1, \dots, k+n-1\}$ , both  $v_i^T M_k v_i = v_i^T A v_i$  and  $v_i^T M_k v_j = 0$ .

At each iteration  $i = k, k+1, \dots, k+n-1$ , Algorithm 3.3.1 obtains the new point by way of exact line search, getting

$$\begin{aligned} x_{i+1} &= x_i - \frac{v_i^T (Ax_i + b)}{v_i^T A v_i} v_i \\ &= x_i - \frac{v_i^T (Ax_k + b + \sum_{j=k}^{i-1} \alpha_j A v_j)}{v_i^T A v_i} v_i \\ &= x_i - \frac{v_i^T (Ax_k + b)}{v_i^T A v_i} v_i - \frac{\sum_{j=k}^{i-1} \alpha_j v_i^T A v_j}{v_i^T A v_i} v_i, \end{aligned}$$

where  $\alpha_j$  is defined by  $\alpha_j = -\frac{v_j^T \nabla f(x_j)}{v_j^T A v_j}$ . Expanding this out over  $n$  consecutive iterations, we obtain

$$x_{k+n} = x_k - \sum_{i=k}^{k+n-1} \frac{v_i^T \nabla f(x_k)}{v_i^T A v_i} v_i - \sum_{i=k}^{k+n-1} \sum_{j=k}^{i-1} \frac{\alpha_j v_i^T A v_j}{v_i^T A v_i} v_i.$$

In particular, this implies

$$\begin{aligned} \|x_{k+n} - x^*\| &\leq \left\| x_k - \sum_{i=k}^{k+n-1} \frac{v_i^T \nabla f(x_k)}{v_i^T A v_i} v_i - x^* \right\| \\ &\quad + \sum_{i=k}^{k+n-1} \sum_{j=k}^{i-1} \left| \frac{v_j^T \nabla f(x_j)}{v_j^T A v_j} \frac{v_i^T A v_j}{v_i^T A v_i} \right|. \end{aligned} \tag{3.3.5}$$

Recall that since  $v_k, \dots, v_{k+n-1}$  are conjugate with respect to  $M_k$  and  $v_i^T A v_i = v_i^T M_k v_i$ , it follows that

$$\left\| x_k - \sum_{i=k}^{k+n-1} \frac{v_i^T \nabla f(x_k)}{v_i^T A v_i} v_i - x^* \right\| = \|x_k - M_k^{-1} \nabla f(x_k) - x^*\|. \tag{3.3.6}$$

Next, recall that since the algorithm is implemented with an exact line search, the objective function is non-increasing at each iteration. Specifically, for all  $j$ ,

$f(x_{j+1}) \leq f(x_j)$ . By Equation 2.2.7, it can be seen that

$$\frac{1}{2}\|x_{j+1} - x^*\|_A^2 = f(x_{j+1}) - f(x^*) \leq f(x_j) - f(x^*) = \frac{1}{2}\|x_j - x^*\|_A^2.$$

This implies that the sequence  $\{x_k\}_{k \geq 0}$  is bounded. Additionally, along with Inequality 2.2.3, this implies that for  $j \geq k$  we have,

$$\begin{aligned} |v_j^T \nabla f(x_j)| &\leq \|v_j\| \|\nabla f(x_j)\| \\ &= \|A(x_j - x^*)\| \\ &\leq \sqrt{\lambda_{\max}(A)} \|x_j - x^*\|_A \\ &\leq \sqrt{\lambda_{\max}(A)} \|x_k - x^*\|_A \\ &\leq \lambda_{\max}(A) \|x_k - x^*\|. \end{aligned}$$

Combining the above inequality, Inequality 2.2.4, Inequality 3.3.3, Inequality 3.3.5 and Equation 3.3.6, we conclude that

$$\|x_{k+n} - x^*\| \leq \|x_k - M_k^{-1} \nabla f(x_k) - x^*\| + \frac{n^2 \lambda_{\max}(A)}{\lambda_{\min}^2(A)} \|x_k - x^*\| \epsilon_k. \quad (3.3.7)$$

Further, observe that since  $\nabla f(x_k) = Ax_k + b = A(x_k - x^*)$ , it follows that

$$\|x_k - M_k^{-1} \nabla f(x_k) - x^*\| = \|(I - M_k^{-1} A)(x_k - x^*)\|. \quad (3.3.8)$$

With the above results, we are ready to prove the superlinear convergence of the algorithm. By Theorem 3.2.3 and Corollary 3.2.4, it follows that  $B_k \rightarrow A$  almost surely, implying  $\epsilon_k \rightarrow 0$  almost surely. Therefore, for all sufficiently large  $k$ , the matrix  $V_k$  is invertible implying that the matrix  $M_k$  is well-defined and that  $M_k \rightarrow A$  almost surely. From that, Equation 3.3.8, and the fact that  $\{x_k - x^*\}_{k \geq 0}$  is bounded, it follows that

$$\|x_k - M_k^{-1} \nabla f(x_k) - x^*\| \rightarrow 0$$

almost surely. Combining this result, the fact that  $\epsilon_k \rightarrow 0$  almost surely and Inequality 3.3.7, it follows that  $\|x_{k+n} - x^*\| \rightarrow 0$  almost surely, proving that the algorithm converges almost surely.

Finally, consider scaling Inequality 3.3.7 by  $\|x_k - x^*\|$ , obtaining

$$\frac{\|x_{k+n} - x^*\|}{\|x_k - x^*\|} \leq \frac{\|x_k - M_k^{-1} \nabla f(x_k) - x^*\|}{\|x_k - x^*\|} + \frac{\frac{n^2 \lambda_{\max}(A)}{\lambda_{\min}^2(A)} \|x_k - x^*\| \epsilon_k}{\|x_k - x^*\|}.$$

Since

$$\frac{\|x_k - M_k^{-1} \nabla f(x_k) - x^*\|}{\|x_k - x^*\|} = \frac{\|(I - M_k^{-1} A)(x_k - x^*)\|}{\|x_k - x^*\|},$$

it follows that the first term converges to zero almost surely since  $M_k \rightarrow A$  almost surely. Further, since  $\epsilon_k \rightarrow 0$  almost surely, the second term converges to zero almost surely. These two facts together imply that

$$\frac{\|x_{k+n} - x^*\|}{\|x_k - x^*\|} \rightarrow 0$$

almost surely: by definition, this means the algorithm is  $n$ -step superlinearly convergent almost surely.  $\square$

In Figure 3.2, we again consider two convex quadratic functions  $\frac{1}{2}x^T A x + b^T x$  where, again,  $A$  is a Hilbert matrix of dimension 7 and  $A = \text{Diag}(1, 7, 7^2, \dots, 7^6)$ , respectively with  $b = [1, 1, \dots, 1]^T$ . The above algorithm was implemented with an exact line search and exact directional second derivatives.

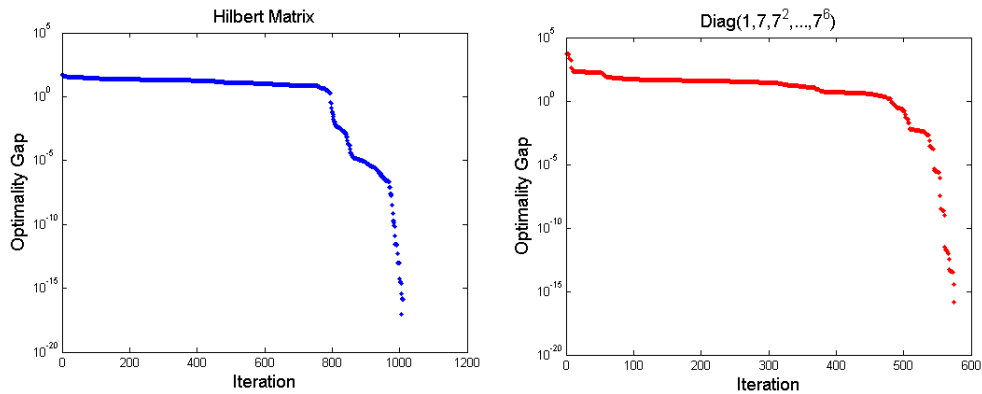


Figure 3.2: Conjugate directions algorithm on convex quadratics



Additionally, in Figure 3.3, we considered two examples of this algorithm on a variant of the Rosenbrock function, given by

$$f(x) = \sum_{i=1}^{n-1} \left[ (1 - x_i)^2 + 10(x_{i+1} - x_i^2)^2 \right],$$

for  $n = 2, 3$ . It was implemented with  $\epsilon = 10^{-4}$ , initial iterate  $[0, 0, \dots, 0]$  and a backtracking line search with initial step size equal to that suggested by the exact quadratic model, estimated via finite differencing, thereby making each iteration require five function evaluations plus any extra cost incurred by the line search. Below, we plot the difference between the present and optimal function against the number of function evaluations required.

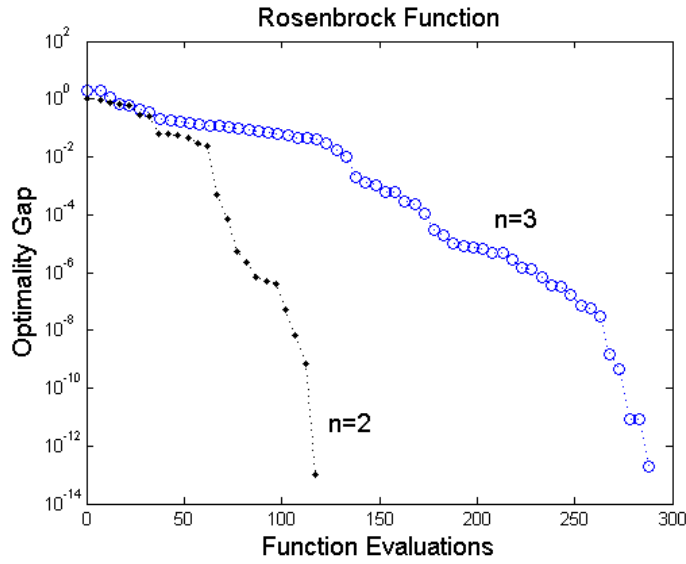


Figure 3.3: Conjugate directions algorithm on Rosenbrock function

For additional comparison, in Figure 3.4, we compared the performance of Algorithm 3.3.1 with one implementation of the Nelder-Mead algorithm, originally presented in [84], by applying both algorithms to the Rosenbrock function with  $n = 10$ . Each algorithm used the initial iterate  $[0, 0, \dots, 0]$  and the conjugate directions algorithm was implemented identically as in the previous example.

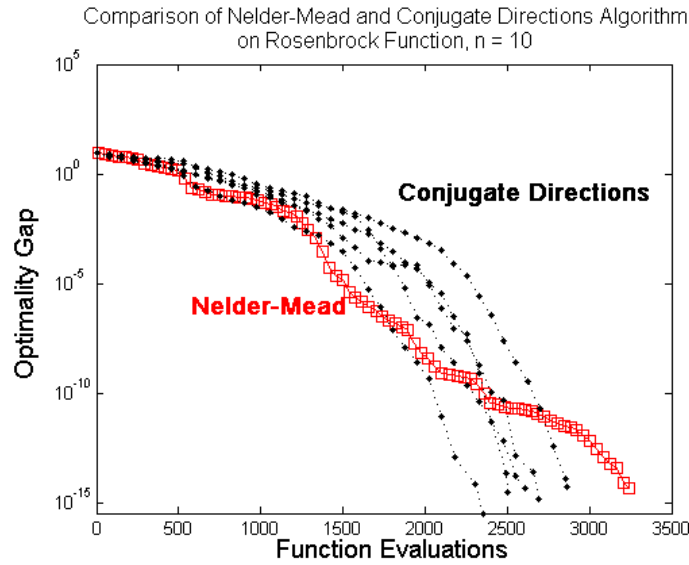


Figure 3.4: Conjugate directions and Nelder-Mead algorithms

### 3.4 Concluding Remarks for Chapter 3

Randomization provides an interesting perspective for a variety of algorithms. Consider the perspective adhered to in this chapter in which our cost measure is the number of function evaluations required, assuming line searches are relatively cheap being a one-dimensional optimization problem, and with derivative information only available through (and well-approximated by) finite differencing. It was then shown in Section 3.1 that random search is comparable to steepest descent. Then, using the Hessian estimation technique introduced in Section 3.2, we demonstrated in Section 3.3 how these techniques can be used to accelerate random search. Finally, we devised a conjugate directions algorithm that incorporates second derivative information without directly requiring gradient information while sharing certain behaviors with more traditional Newton-like methods.

We make no claim that the conceptual techniques described above, in their pure form, are competitive with already-known derivative-based or derivative-free algorithms. We simply intend to illustrate how incorporating randomization provides a novel approach to the design of algorithms, even in very simple optimization schemes, suggesting that it may deserve further consideration. Note that all the algorithms considered in this chapter, at each iteration, require only directional derivative or directional second-order information, creating a connection between the realms of derivative-free and derivative-based algorithms when this derivative information is well-approximated by finite differencing.

## CHAPTER 4

### RANDOMIZED METHODS FOR LINEAR CONSTRAINTS

#### 4.1 Introduction

In Chapter 3, several examples of randomized algorithms were examined from what is typically viewed as a “derivative-free optimization perspective,” where computational costs are loosely measured in the number of function evaluations performed. Within that specific framework, randomized algorithmic schemes were compared with derivative-based approaches like the classical steepest descent algorithm. However, due to the focus on function evaluations as the algorithmic cost, the related costs of arithmetic operations—such as matrix-vector multiplication and matrix inversion—were ignored.

The underlying problem being considered was that of minimizing a convex quadratic function, which we will now assume to take the form  $f(x) = \frac{1}{2}x^T Ax - b^T x$ , where  $A$  is a positive definite matrix. An equivalent linear-algebraic formulation for solving this problem is to focus on finding a solution to system  $\nabla f(x) = Ax - b = 0$ , leading to a solution that satisfies the first-order necessary optimality conditions (which, in this case, are also sufficient conditions).

This chapter will begin with an examination of this problem from a perspective where the costs of linear-algebraic operations are themselves a primary concern. After seeing the role of randomized algorithms in the context of solving positive-definite linear systems, generalizations will be considered to indefinite systems, least squares problems, linear inequality systems and, finally, convex

feasibility problems.

In this setting, the motivation for considering randomized algorithms is two-fold. Ideally, one hope is that by using randomization as an algorithmic tool, certain randomization schemes will lead to improved computational performance. However, there is also an interest in considering randomization as an analytic tool in its own right. By this, we mean to show that randomization can be used to obtain meaningful, quantifiable convergence rates that demonstrate the interplay between the conditioning of a problem and its theoretical performance. This is in contrast with deterministic variants of many of the algorithms under consideration, for which convergence behavior is known, but is either not quantifiable or for which the known convergence rate has a tenuous association with conditioning information for the problem input.

## 4.2 Randomized Coordinate Descent

### 4.2.1 The Basic Result: Positive Semi-Definite Systems

To begin, return to the problem of solving a linear system of the form  $Ax = b$ , with  $A$  being an  $n$ -by- $n$  positive-definite matrix and associated solution  $x^* = A^{-1}b$ , and the equivalent problem of minimizing the strictly convex quadratic function

$$f(x) = \frac{1}{2}x^T Ax - b^T x.$$

Proceeding in a manner similar to Section 3.1, suppose our current iterate is  $x$  and we obtain a new iterate  $x_+$  by performing an exact line search in the nonzero

direction  $d$ : that is,  $x_+$  is the solution to  $\min_{x+\mathbb{R}d} f$ . This gives us

$$x_+ = x - \frac{(Ax - b)^T d}{d^T A d} d$$

and, by Equation 2.2.7

$$f(x_+) - f(x^*) = \frac{1}{2} \|x_+ - x^*\|_A^2 = \frac{1}{2} \|x - x^*\|_A^2 - \frac{((Ax - b)^T d)^2}{2d^T A d}. \quad (4.2.1)$$

Given the motivation of reducing the number of arithmetic operations required, one natural choice for a set of easily-computable search directions is to choose  $d$  from the set of coordinate directions,  $\{e_1, \dots, e_n\}$ . Note that, when using search direction  $e_i$ , we can compute the new point

$$x_+ = x + \frac{b_i - a_i^T x}{a_{ii}} e_i$$

so that each iteration does not require a matrix-vector product, instead using only  $2n + 2$  arithmetic operations. If the search direction is chosen at each iteration by successively cycling through the set of coordinate directions, then the algorithm is known to be linearly convergent but with a rate not easily expressible in terms of typical matrix quantities (see [48] or [93]). However, by choosing a coordinate direction as a search direction randomly according to an appropriate probability distribution, we can obtain a convergence rate in terms of the scaled or relative condition numbers. In considering the following algorithm, we will weaken our assumptions and merely require the matrix  $A$  to be positive semidefinite.

**Algorithm 4.2.2** Consider an  $n$ -by- $n$  positive semidefinite system  $Ax = b$  with  $A \neq 0$  and let  $x_0 \in \mathbb{R}^n$  be an arbitrary starting point. For  $j = 0, 1, 2, \dots$ , compute

$$x_{j+1} = x_j + \frac{b_i - a_i^T x_j}{a_{ii}} e_i$$

where, at each iteration  $j$ , the index  $i$  is chosen independently at random from the set  $\{1, \dots, n\}$ , with distribution

$$P\{i = k\} = \frac{a_{kk}}{\text{tr } A}.$$

Notice in the algorithm that the matrix  $A$  may be singular, but at each iteration, it follows that  $a_{ii} > 0$  almost surely for the randomly chosen index  $i$ . If  $A$  is merely positive semidefinite, solutions of the system  $Ax = b$  coincide with minimizers of the function  $f$ , and consistency of the system is equivalent to  $f$  being bounded below. We now have the following result.

**Theorem 4.2.3** *Consider a consistent positive-semidefinite system  $Ax = b$ , and define the corresponding objective and error by*

$$\begin{aligned} f(x) &= \frac{1}{2}x^T Ax - b^T x \\ \delta(x) &= f(x) - \min f. \end{aligned}$$

*Then Algorithm 4.2.2 satisfies, for each iteration  $j = 0, 1, 2, \dots$ ,*

$$\mathbf{E}[\delta(x_{j+1}) \mid x_j] \leq \left(1 - \frac{\lambda(A)}{\text{tr } A}\right) \delta(x_j).$$

*In particular, if  $A$  is positive-definite and  $x^* = A^{-1}b$ , we have that the sequence of iterates generated by Algorithm 4.2.2 is linearly convergent in expectation with respect to  $\|\cdot\|_A$ , satisfying the equivalent property*

$$\mathbf{E}[\|x_{j+1} - x^*\|_A^2 \mid x_j] \leq \left(1 - \frac{1}{\|A^{-1}\|_2 \text{tr } A}\right) \|x_j - x^*\|_A^2.$$

*Hence, the expected reduction in the squared error  $\|x_j - x^*\|_A^2$  is at least a factor*

$$1 - \frac{1}{\sqrt{n}\kappa(A)} \leq 1 - \frac{1}{n\kappa(A)}$$

*at each iteration.*

**Proof** Note that if coordinate direction  $e_i$  is chosen during iteration  $j$ , then Equation 4.2.1 shows

$$f(x_{j+1}) = f(x_j) - \frac{(b_i - a_i^T x_j)^2}{2a_{ii}}.$$

Hence, using

$$\mathbf{E}[f(x_{j+1}) \mid x_j] = f(x_j) - \sum_{i=1}^n \frac{a_{ii}}{\text{tr}(A)} \frac{(b_i - a_i^T x_j)^2}{2a_{ii}},$$

we deduce

$$\mathbf{E}[f(x_{j+1}) \mid x_j] = f(x_j) - \frac{1}{2\text{tr} A} \|Ax_j - b\|^2. \quad (4.2.4)$$

Using Inequality 2.2.5 and Equation 2.2.7, we easily verify

$$\frac{1}{2} \|Ax_j - b\|^2 \geq \underline{\lambda}(A) \delta(x_j),$$

and the first result follows. Applying Equation 2.2.7 provides the second result.

The final result comes from applying Inequalities 2.2.1 and 2.2.6.  $\square$

Consider for a moment the case when the system  $Ax = b$  is inconsistent. In that case, the quantity  $\|Ax - b\|$  is bounded below by some strictly positive constant. Equation 4.2.4 therefore implies the existence of a constant  $\epsilon > 0$  such that

$$\mathbf{E}[f(x_{j+1}) \mid x_j] \leq f(x_j) - \epsilon, \text{ for all } j.$$

We know  $f(x_{j+1}) \leq f(x_j)$ . The description of the algorithm implies that, at each iteration, the probability that we observe  $f(x_{j+1}) \leq f(x_j) - \epsilon$  is at least some fixed positive constant. Hence  $f(x_j) \downarrow -\infty$  almost surely.

The simple idea behind the proof of Theorem 4.2.3 is the main engine driving many of the remaining results in this chapter. Fundamentally, the idea is to choose a probability distribution so that the expected distance to the solution from the new iterate is the distance to the solution from the old iterate minus



some multiple of a residual. Then, using some type of error bound to bound the distance to a solution in terms of the residual, we obtain expected linear convergence of the algorithm.

## 4.2.2 The General Result: Positive Semi-Definite Systems

Although the mathematical simplicity induced by randomization in the proof of Theorem 4.2.3 will be re-iterated throughout this chapter, it is of independent mathematical interest that an identical result can be obtained for a more general algorithm. From a computational standpoint, coordinate descent algorithms are appealing in that each iteration does not require matrix-vector multiplications, instead using only  $O(n)$  arithmetic operations. However, the convergence rate shown in Theorem 4.2.3 only involves the eigenvalues of the matrix  $A$ , suggesting a degree of rotational invariance inherent in that result. Before proving the next result, some additional definitions will be provided.

A set  $V \subseteq \mathbb{R}^n$  is an **orthonormal basis** if  $|V| = n$  and for all  $x, y \in V$  such that  $x \neq y$ , it follows that  $\|x\| = 1$  and  $\langle x, y \rangle = 0$ . Let  $I$  be an index set with a corresponding probability measure  $P_I$ , let  $\{D_i : i \in I\}$  be a collection of orthonormal bases indexed by  $I$ . Using this notation, we can define the following algorithm.

**Algorithm 4.2.5** *Let  $I$  be a measurable index set and  $\{D_i : i \in I\}$  be a collection of orthonormal bases. Consider the linear system  $Ax = b$ , with  $A$  positive semi-definite, and let  $x_0$  be an arbitrary vector. For  $j = 0, 1, 2, \dots$ , compute*

$$x_{j+1} = x_j + \frac{v_j^T(b - Ax_j)}{v_j^T A v_j} v_j$$

where, at each iteration  $j$ ,  $v_j$  is chosen independently at random according to the probability distribution defined by

$$\gamma(dv) := \frac{v^T A v}{\text{tr}(A)} P_I(\{i \in I : dv \in D_i\}).$$

One interpretation of this distribution is provided by thinking of the algorithm as sampling  $i \in I$  according to  $P_I(\cdot)$  and then, conditional on  $i$ , choosing  $v \in D_i$  with probability  $\frac{v^T A v}{\text{tr}(A)}$ . By this argument, we obtain a distribution  $P(dv)$  for  $v$  by

$$P(dv) = \int_I P_I(i) \left[ \frac{v^T A v}{\text{tr}(A)} 1_{v \in D_i} \right] di = \frac{v^T A v}{\text{tr}(A)} \int_I P_I(i) 1_{v \in D_i} di = \gamma(dv),$$

where  $1_S(x)$  is 1 if  $x \in S$  and 0 otherwise. In particular, according to the distribution  $\gamma(\cdot)$ , the search direction  $v_j \in \cup_i D_i$  almost surely. As a special case, note that by choosing  $I = \{1\}$  and  $D_1 = \{e_1, \dots, e_n\}$ , we obtain Algorithm 4.2.2. From this, we would expect a convergence result similar to that given in Theorem 4.2.3. As shown in the following theorem, this is precisely the case.

**Theorem 4.2.6** *Consider a consistent positive semi-definite system  $Ax = b$ . Then the conclusions of Theorem 4.2.3 are valid for Algorithm 4.2.5 as well.*

**Proof** First, suppose that  $D_i$  is an orthonormal basis. Then it follows that

$$\int_{D_i} [(b - Ax)^T y]^2 dy = \sum_{y \in D_i} [(b - Ax)^T y]^2 = \|Ax - b\|^2. \quad (4.2.7)$$

Now, suppose that at iteration  $j$ , search direction  $v \in \mathbf{S}$  is chosen. Then the new iterate  $x_{j+1}$  satisfies

$$f(x_{j+1}) = f(x_j) - \frac{[(b - Ax_j)^T v]^2}{2v^T A v}.$$

Taking the expectation with respect to the stated probability distribution gives

$$\begin{aligned}
\mathbf{E}[f(x_{j+1}) | x_j] &= f(x_j) - \mathbf{E}\left[\frac{[(b - Ax_j)^T y]^2}{2y^T Ay} \mid x_j\right] \\
&= f(x_j) - \frac{1}{2} \int_{\cup_i D_i} \frac{[(b - Ax_j)^T y]^2}{y^T Ay} \frac{y^T Ay}{\text{tr}(A)} P_I(\{i \in I : y \in D_i\}) dy \\
&= f(x_j) - \frac{1}{2\text{tr}(A)} \int_{\cup_i D_i} [(b - Ax_j)^T y]^2 P_I(\{i \in I : y \in D_i\}) dy \\
&= f(x_j) - \frac{1}{2\text{tr}(A)} \int_I \left[ \int_{D_i} [(b - Ax_j)^T y]^2 dy \right] P_I(di) \\
&= f(x_j) - \frac{\|Ax_j - b\|^2}{2\text{tr}(A)} \int_I P_I(di) \quad (\text{Equation 4.2.7}) \\
&= f(x_j) - \frac{\|Ax_j - b\|^2}{2\text{tr}(A)}.
\end{aligned}$$

By noting that this is exactly Equation 4.2.4 in the proof of Theorem 4.2.3, the remainder of the proof follows.  $\square$

### 4.2.3 General Linear Systems

Now let us consider the more general problem of finding a solution to a linear system  $Ax = b$  where  $A$  is an  $m \times n$ . Since the system might be inconsistent, we seek a “least squares solution” that minimizes the function  $\|Ax - b\|^2$ . Without loss of generality, assume that  $A_i \neq 0$  for  $i = 1, \dots, n$ . It can then be verified that the minimizers are exactly the solutions of the positive-semidefinite system  $A^T Ax = A^T b$ , to which we could easily apply the previous algorithm; however, we wish to avoid computing the new matrix  $A^T A$  explicitly. Instead, we can proceed as follows.

**Algorithm 4.2.8** Consider a linear system  $Ax = b$  for a nonzero  $m$ -by- $n$  matrix  $A$ . Let

$x_0 \in \mathbb{R}^n$  be an arbitrary initial point and let  $r_0 = b - Ax_0$  be the initial residual. For each  $j = 0, 1, \dots$ , compute

$$\begin{aligned}\alpha_j &= \frac{A_i^T r_j}{\|A_i\|^2} \\ x_{j+1} &= x_j + \alpha_j e_i \\ r_{j+1} &= r_j - \alpha_j A_i,\end{aligned}$$

where, at each iteration  $j$ , the index  $i$  is chosen independently at random from the set  $\{1, \dots, n\}$ , with distribution

$$P\{i = k\} = \frac{\|A_k\|^2}{\|A\|_F^2} \quad (k = 1, 2, \dots, n).$$

Note that the step size at each iteration can be obtained by directly minimizing the residual in the respective coordinate direction. However, the algorithm can also be viewed as the application of the algorithm for positive definite systems on the system of normal equations,  $A^T A x = A^T b$ , without actually requiring the computation of the matrix  $A^T A$ . Given the motivation of directly minimizing the residual, we would expect that Algorithm 4.2.8 would converge to a least squares solution, even in the case where the underlying system is inconsistent. The next result shows that this is, in fact, the case.

**Theorem 4.2.9** Consider any linear system  $Ax = b$ , where the matrix  $A$  is nonzero. Define the residual and the error by

$$\begin{aligned}f(x) &= \frac{1}{2} \|Ax - b\|^2 \\ \delta(x) &= f(x) - \min f.\end{aligned}$$

Then Algorithm 4.2.8 satisfies, for each iteration  $j = 0, 1, 2, \dots$ ,

$$\mathbf{E}[\delta(x_{j+1}) \mid x_j] \leq \left(1 - \frac{\lambda(A^T A)}{\|A\|_F^2}\right) \delta(x_j).$$

In particular, if  $A$  has full column rank, then the sequence of iterates generated by Algorithm 4.2.8 is linearly convergent in expectation with respect to  $\|\cdot\|_{A^T A}$ , satisfying

$$\mathbf{E}[\|x_{j+1} - \hat{x}\|_{A^T A}^2 \mid x_j] \leq \left(1 - \frac{1}{\kappa(A)^2}\right) \|x_j - \hat{x}\|_{A^T A}^2$$

where  $\hat{x} = (A^T A)^{-1} A^T b$  is the unique least-squares solution.

**Proof** It is easy to verify, by induction on  $j$ , that the iterates  $x_j$  are exactly the same as the iterates generated by Algorithm 4.2.2, when applied to the positive semi-definite system  $A^T A x = A^T b$ , and furthermore that the residuals satisfy  $r_j = b - A x_j$  for all  $j = 0, 1, 2, \dots$ . Hence, the results follow directly by Theorem 4.2.3.  $\square$

By the coordinate descent nature of this algorithm, once we have computed the initial residual  $r_0$  and column norms  $\{\|A_i\|^2\}_{i=1}^n$ , we can perform each iteration in  $O(m)$  time. Specifically, this new iteration takes  $4m + 1$  arithmetic operations, compared with  $2n + 2$  for the positive-definite case.

For a computational example, we apply Algorithm 4.2.8 to random  $500 \times n$  matrices where each element of  $A$  and  $b$  is an independent standard Gaussian random variable and we let  $n$  take values 50, 100, 150 and 200. The results are shown in Figure 4.1. Note that in these examples, the theoretical bound provided by Theorem 4.2.9 predicts the actual behavior of the algorithm reasonably well.

### 4.3 Randomized Iterated Projections

Iterated projection algorithms share some important characteristics with coordinate descent algorithms. Both are well studied and much convergence theory

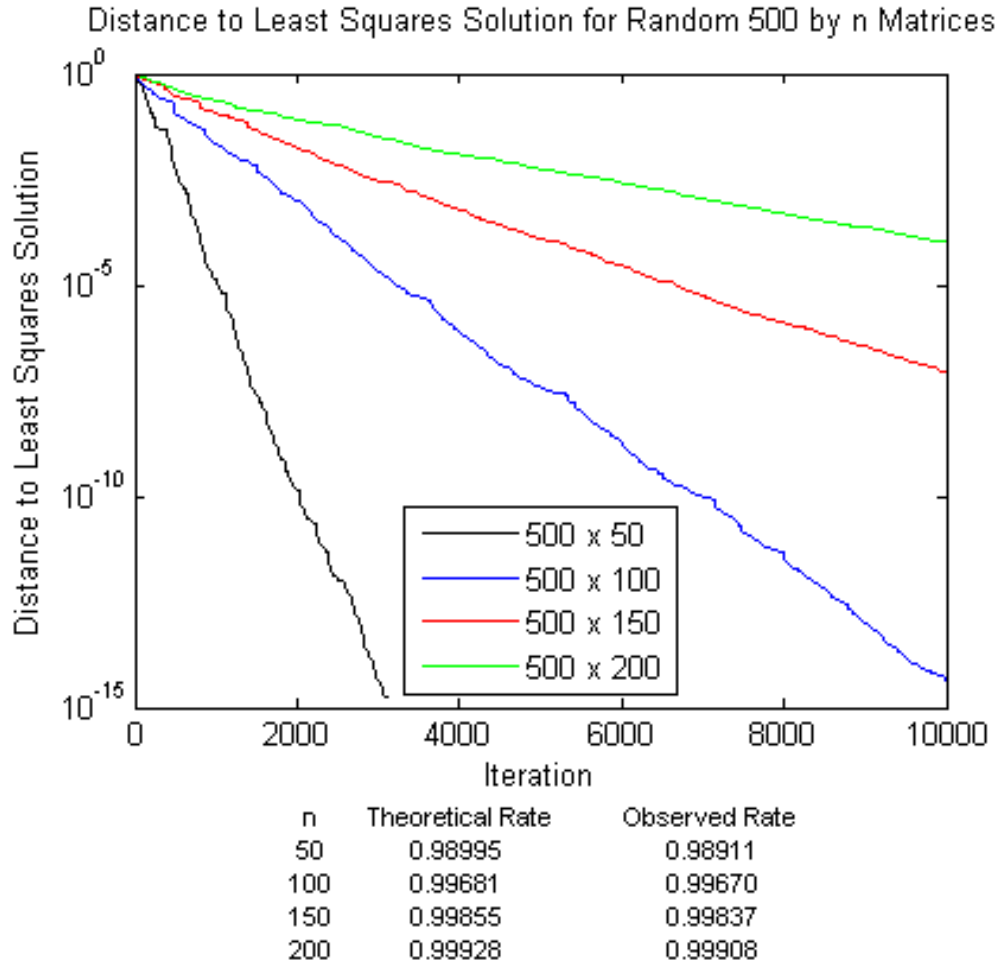


Figure 4.1: Randomized coordinate descent algorithm for least squares problems

exists; a comprehensive overview on iterated projections can be found in [30]. Randomized iterated projection methods have also been considered by many authors in a variety of mathematical settings. Convergence results for very general frameworks can be found in [3], [20], [36], and [6], among others. Results on randomized algorithms for convex feasibility problems in  $\mathbb{R}^n$  have been further developed by [90] and [2], for example, including convergence theory for infeasible systems. However, even for linear systems of equations, standard

developments do not provide bounds on convergence rates in terms of natural numerical quantities. For example, the convergence rates given in [30] and [44] for a cyclic projections algorithm depend on the angles between certain intersections of affine spaces. By contrast, in the recent paper [109], Strohmer and Vershynin obtained a natural convergence rate via the following randomized iterated projection algorithm, which also provided the motivation for our work in this chapter.

**Algorithm 4.3.1** Consider a linear system  $Ax = b$  for a nonzero  $m$ -by- $n$  matrix  $A$ . Let  $x_0 \in \mathbb{R}^n$  be an arbitrary initial point. For each  $j = 0, 1, 2, \dots$ , compute

$$x_{j+1} = x_j - \frac{a_i^T x_j - b_i}{\|a_i\|^2} a_i$$

where, at each iteration  $j$ , the index  $i$  is chosen independently at random from the set  $\{1, \dots, m\}$ , with distribution

$$P\{i = k\} = \frac{\|a_k\|^2}{\|A\|_F^2} \quad (k = 1, 2, \dots, m).$$

Notice that the new iterate  $x_{j+1}$  is simply the orthogonal projection of the old iterate  $x_j$  onto the hyperplane  $\{x : a_i^T x = b_i\}$ . At first sight, the choice of probability distribution may seem curious, since we could rescale the equations arbitrarily without having any impact on the projection operations. However, following [109], we emphasize that the aim is to understand linear convergence rates in terms of *linear-algebraic* condition measures associated with the original system, rather than in terms of *geometric* notions associated with the hyperplanes. More generally, the use of row and column norms in the design analysis of randomized algorithms appears in a variety of applications for matrix problems; some examples include low-rank matrix approximation in [43],  $l_2$ -regression in [35]

and matrix multiplication in [34]. In particular, this randomized algorithm has the following behavior.

**Theorem 4.3.2 (Strohmer-Vershynin, [109])** *Given any matrix  $A$  with full column rank, suppose the linear system  $Ax = b$  has solution  $x^*$ . Then the sequence of iterates generated by Algorithm 4.3.1 is linearly convergent in expectation, satisfying*

$$\mathbf{E}[\|x_{j+1} - x^*\|_2^2 \mid x_j] \leq \left(1 - \frac{1}{\kappa(A)^2}\right) \|x_j - x^*\|_2^2.$$

Several authors have observed that randomized iterated projection schemes often outperform deterministic variants for specially structured problems (see [54], [41] and the references therein), though Theorem 4.3.2 is the first known appearance of a provable convergence rate. In particular, the authors of [109] observe that Algorithm 4.3.1 appears to substantially outperform both uniformly randomized and deterministic variants for the problem of reconstructing bandlimited signals with non-uniform sampling, consistent with prior observations. One possible explanation for this behavior, conjectured in [22] with regards to irregular sampling problems, involves the computational observation that if we considered an algorithm that projects onto the hyperplane which provides the greatest reduction in residual error,  $\|Ax - b\|$ , then such an algorithm tends to project onto higher norm rows with greater frequency. Further discussion about Algorithm 4.3.1 with regards to this problem can be found in [23].

We seek a way of generalizing the above algorithm and convergence result to more general systems of linear inequalities, of the form

$$\begin{cases} a_i^T x \leq b_i & (i \in I_{\leq}) \\ a_i^T x = b_i & (i \in I_{=}), \end{cases} \quad (4.3.3)$$



where the disjoint index sets  $I_{\leq}$  and  $I_{=}$  partition the set  $\{1, 2, \dots, m\}$ . To do so, staying with the techniques of the previous section, we need a corresponding error bound for a system of linear inequalities. A starting point for this subject is a result by Hoffman in [58].

**Theorem 4.3.4 (Hoffman)** *For any right-hand side vector  $b \in \mathbb{R}^m$ , let  $S_b$  be the set of feasible solutions of the linear system (4.3.3). Then there exists a constant  $L$ , independent of  $b$ , with the following property:*

$$x \in \mathbb{R}^n \text{ and } S_b \neq \emptyset \Rightarrow d(x, S_b) \leq L \|e(Ax - b)\|, \quad (4.3.5)$$

where the function  $e: \mathbb{R}^m \rightarrow \mathbb{R}^m$  is defined by

$$e(y)_i = \begin{cases} y_i^+ & (i \in I_{\leq}) \\ y_i & (i \in I_{=}). \end{cases}$$

In the above result, each component of the vector  $e(Ax - b)$  indicates the error in the corresponding inequality or equation. In particular  $e(Ax - b) = 0$  if and only if  $x \in S_b$ . Thus Hoffman's result provides a linear bound for the distance from a trial point  $x$  to the feasible region in terms of the size of the "a posteriori error" associated with  $x$ .

We call the minimum constant  $L$  such that property (4.3.5) holds the *Hoffman constant* for the system (4.3.3). Several authors give geometric or algebraic meaning to this constant, or exact expressions for it, including [52], [85], [71], [57], [114], the survey [87], and some generalizations in [21]. In the case of linear equations (that is,  $I_{\leq} = \emptyset$ ), an easy calculation using the singular value decomposition shows that the Hoffman constant is just the reciprocal of the smallest nonzero singular value of the matrix  $A$ , and hence equals  $\|A^{-1}\|_2$  when  $A$  has full column rank.

For the problem of finding a solution to a system of linear inequalities, we consider a randomized algorithm generalizing Algorithm 4.3.1.

**Algorithm 4.3.6** Consider the system of inequalities described by 4.3.3 and let  $x_0$  be an arbitrary initial point. For each  $j = 0, 1, \dots$ , compute

$$\begin{aligned}\beta_j &= \begin{cases} (a_i^T x_j - b_i)^+ & (i \in I_{\leq}) \\ a_i^T x_j - b_i & (i \in I_{=}) \end{cases} \\ x_{j+1} &= x_j - \frac{\beta_j}{\|a_i\|^2} a_i\end{aligned}$$

where, at each iteration  $j$ , the index  $i$  is chosen independently at random from the set  $\{1, \dots, m\}$ , with distribution

$$P\{i = k\} = \frac{\|a_k\|^2}{\|A\|_F^2} \quad (k = 1, 2, \dots, m).$$

In the above algorithm, notice  $\beta_j = e(Ax_j - b)_i$  and that  $x_{j+1}$  is just the orthogonal projection onto the halfspace or hyperplane defined by the constraint with index  $i$ . We can now generalize Theorem 4.3.2 as follows.

**Theorem 4.3.7** Suppose the system (4.3.3) has nonempty feasible region  $S$ . Then the sequence of iterates generated by Algorithm 4.3.6 is linearly convergent in expectation, satisfying

$$\mathbf{E}[d(x_{j+1}, S)^2 \mid x_j] \leq \left(1 - \frac{1}{L^2 \|A\|_F^2}\right) d(x_j, S)^2$$

where  $L$  is the Hoffman constant.

**Proof** Note that if the index  $i$  is chosen during iteration  $j$ , then it follows that

$$\begin{aligned}\|x_{j+1} - P_S(x_{j+1})\|^2 &\leq \|x_{j+1} - P_S(x_j)\|^2 \\ &= \left\| x_j - \frac{e(Ax_j - b)_i}{\|a_i\|^2} a_i - P_S(x_j) \right\|^2 \\ &= \|x_j - P_S(x_j)\|^2 + \frac{e(Ax_j - b)_i^2}{\|a_i\|^2} - 2 \frac{e(Ax_j - b)_i}{\|a_i\|^2} a_i^T (x_j - P_S(x_j)).\end{aligned}$$

Note  $P_S(x_j) \in S$ . Hence if  $i \in I_{\leq}$ , then  $a_i^T P_S(x_j) \leq b_i$ , and  $e(Ax_j - b)_i \geq 0$ , so

$$e(Ax_j - b)_i a_i^T (x_j - P_S(x_j)) \geq e(Ax_j - b)_i (a_i^T x_j - b_i) = e(Ax_j - b)_i^2.$$

On the other hand, if  $i \in I_{=}$ , then  $a_i^T P_S(x_j) = b_i$ , so

$$e(Ax_j - b)_i a_i^T (x_j - P_S(x_j)) = e(Ax_j - b)_i (a_i^T x_j - b_i) = e(Ax_j - b)_i^2.$$

Putting these two cases together with the previous inequality shows

$$d(x_{j+1}, S)^2 \leq d(x_j, S)^2 - \frac{e(Ax_j - b)_i^2}{\|a_i\|^2}.$$

Taking the expectation with respect to the specified probability distribution, it follows that

$$\mathbf{E}[d(x_{j+1}, S)^2 \mid x_j] \leq d(x_j, S)^2 - \frac{\|e(Ax_j - b)\|^2}{\|A\|_F^2} \quad (4.3.8)$$

and the result now follows by the Hoffman bound.  $\square$

Since Hoffman's bound is not independent of the scaling of the matrix  $A$ , it is not surprising that a normalizing constant like  $\|A\|_F^2$  term appears in the result.

It's worth noting that Theorem 4.3.7 allows us to remove the full column rank assumption in the case of a linear equality system, providing a similar convergence rate as in Theorem 4.3.2.

**Corollary 4.3.9** *Suppose the linear system  $Ax = b$ , with  $A \neq 0$ , has a non-empty solution set  $S$ . Then the sequence of iterates generated by Algorithm 4.3.1 is linearly convergent in expectation, satisfying*

$$\mathbf{E}[d(x_{j+1}, S)^2 \mid x_j] \leq \left(1 - \frac{\sigma(A)^2}{\|A\|_F^2}\right) d(x_j, S)^2$$

**Proof** The result follows from the fact that the Hoffman constant for a consistent linear system is the reciprocal of the smallest non-zero singular value.  $\square$

For a computational example, we consider linear inequality systems  $Ax \leq b$  where the elements of  $A$  are independent standard Gaussian random variables and  $b$  is chosen so that the resulting system has a non-empty interior (specifically, letting  $d$  be a vector of independent, standard Gaussian random variables,  $b = Ad + .01e$  where  $e = [1, 1, \dots, 1]^T$ ). We consider matrices  $A$  which are  $500 \times n$  and let  $n$  take values 50, 100, 150 and 200. We then apply Algorithm 4.3.6 to these problems and observe the following computational results in Figure 4.2.

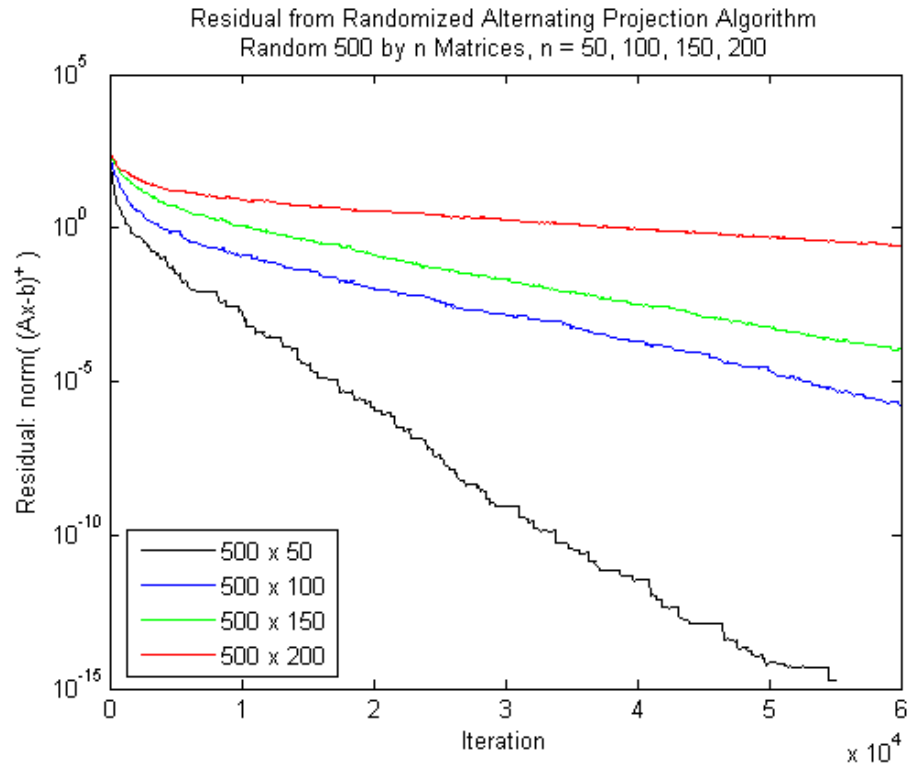


Figure 4.2: Randomized alternating projection algorithm for linear inequalities

Another natural conditioning measure for linear inequality systems is the distance to infeasibility, defined by Renegar in [96], and shown in [98] to govern the convergence rate of interior point methods for linear programming. It is interesting, therefore, from a theoretical perspective, to obtain a linear convergence rate for iterated projection algorithms in terms of this condition measure as well. For simplicity, we concentrate on the inequality case. To begin, let us recall the following results.

**Definition 4.3.10 ([96])** *The **distance to infeasibility** for the system  $Ax \leq b$  is the number*

$$\mu = \inf \left\{ \max\{\|\Delta A\|_2, \|\Delta b\|\} : (A + \Delta A)x \leq b + \Delta b \text{ is infeasible} \right\}.$$

**Theorem 4.3.11 (Renegar, [96], Thm 1.1)** *Consider the system  $Ax \leq b$ . Suppose the distance to infeasibility  $\mu > 0$ . Then there exists a point  $\hat{x}$  in the feasible region  $S$  satisfying  $\|\hat{x}\| \leq \|b\|/\mu$ . Furthermore, any point  $x \in \mathbb{R}^n$  satisfies the inequality*

$$d(x, S) \leq \frac{\max\{1, \|x\|\}}{\mu} \|(Ax - b)^+\|.$$

Using this, we can bound the linear convergence rate for the Algorithm 4.3.6 in terms of the distance to infeasibility, as follows. As before, let  $S = \{x : Ax \leq b\}$ . Suppose we start Algorithm 4.3.6 at the initial point  $x_0 = 0$  and notice that  $\|x_j - \hat{x}\|$  is nonincreasing in  $j$  by Inequality 2.3.7. Applying Theorem 4.3.11, we see that for all  $j = 1, 2, \dots$ ,

$$\|x_j\| \leq \|\hat{x}\| + \|x_j - \hat{x}\| \leq \|\hat{x}\| + \|x_0 - \hat{x}\| \leq \frac{2\|b\|}{\mu},$$

so

$$d(x_j, S) \leq \max \left\{ \frac{1}{\mu}, \frac{2\|b\|}{\mu^2} \right\} \|(Ax_j - b)^+\|.$$

Using this inequality in place of Hoffman's bound in the proof of Theorem 4.3.7 gives

$$E[d(x_{j+1}, S)^2 \mid x_j] \leq \left[ 1 - \frac{1}{\|A\|_F^2 (\max\{\frac{1}{\mu}, \frac{2\|b\|}{\mu^2}\})^2} \right] d(x_j, S)^2.$$

Although this bound may not be the best possible (and, in fact, it may not be as good as the bound provided in Theorem 4.3.7), this result simply emphasizes a relationship between algorithm speed and conditioning measures that appears naturally in other contexts. In fact, as shown in [32, Sec. 4], the distance to infeasibility from Definition 4.3.10 for a linear-conic system can be expressed in the framework of metric regularity, as defined in Section 2.3.2. In the next section, we proceed with these ideas along that framework.

## 4.4 Metric Regularity and Local Convergence

The previous section concerned global rates of linear convergence. If instead we are interested in *local* rates, we can re-examine a generalization of our problem through an alternative perspective of set-valued mappings. Consider a set-valued mapping  $\Phi : \mathbb{E} \rightrightarrows \mathbb{Y}$  and the problem of solving the associated constraint system of the form  $b \in \Phi(x)$  for the unknown vector  $x$ . For example, finding a feasible solution to  $Ax \leq b$  is equivalent to finding an  $x$  such that

$$b \in Ax + \mathbb{R}_+^m. \tag{4.4.1}$$

In this setting, taking  $\Phi(x) = Ax + \mathbb{R}_+^m$ , it follows that  $d(b, \Phi(x)) = \|(Ax - b)^+\|$ . Hence, if the linear inequality system is feasible, the Hoffman bound of Theorem 4.3.4 provides the existence of a constant  $\gamma$  such that

$$d(x, \Phi^{-1}(b)) \leq \gamma d(b, \Phi(x)).$$

This suggests metric regularity or metric subregularity as a tool for generalizing the results of Section 4.3 to constraint systems, at the expense of those results now holding only locally instead of globally.

We wish to consider how the modulus of (sub)regularity of  $\Phi$  affects the convergence rate of iterated projection algorithms. We remark that linear convergence for iterated projection methods on convex sets has been very widely studied. For example, for two closed, convex sets, regularity conditions for linear convergence were proved in [7], generalizing results found in [50]. Broad surveys of the topic for multiple sets can be found in [30] and [8]. Our aim here is to observe, by analogy with previous sections, how randomization makes the linear convergence rate easy to interpret in terms of metric regularity. Under appropriate metric subregularity assumptions, the following local convergence result is obtained.

**Theorem 4.4.2** *Let  $S_1, \dots, S_m$  be closed, convex sets and suppose the set-valued mapping  $\Phi$  given by Equation 2.3.26 is metrically subregular at  $\bar{x}$  for 0 with subregularity modulus  $\text{Subreg } \Phi(\bar{x}|0)$ . Define  $S = \bigcap_i S_i$ , let  $\bar{\gamma} > \text{Subreg } \Phi(\bar{x}|0)$  and let  $x_0$  be any initial point sufficiently close to  $\bar{x}$ . Further, suppose that  $x_{j+1} = P_{S_i}(x_j)$  with probability  $\frac{1}{m}$  for  $i = 1, \dots, m$ . Then the sequence  $\{x_j\}_{j \geq 0}$  is linearly convergent in expectation, satisfying*

$$\mathbf{E}[d(x_{j+1}, S)^2 \mid x_j] \leq \left(1 - \frac{1}{m\bar{\gamma}^2}\right)d(x_j, S)^2.$$

**Proof** First, note that by Inequality 2.3.7, the distance  $\|x_j - \bar{x}\|$  is nonincreasing in  $j$ . Hence if  $x_0$  is sufficiently close to  $\bar{x}$  so that Inequality 2.3.22 holds with constant  $\bar{\gamma}$ , then  $x_j$  is as well for all  $j \geq 0$ . Then, again using Inequality 2.3.7

(applied to the set  $S_i$ ), we have, for all points  $x \in S \subset S_i$ ,

$$\|x_j - x\|^2 - \|x_j - P_{S_i}(x_j)\|^2 \geq \|P_{S_i}(x_j) - x\|^2.$$

Taking the minimum over  $x \in S$ , we deduce

$$d(x_j, S)^2 - \|x_j - P_{S_i}(x_j)\|^2 \geq d(P_{S_i}(x_j), S)^2.$$

Hence

$$\begin{aligned} \mathbf{E}[d(x_{j+1}, S)^2 \mid x_j] &= \frac{1}{m} \sum_{i=1}^m d(P_{S_i}(x_j), S)^2 \\ &\leq \frac{1}{m} \sum_{i=1}^m [d(x_j, S)^2 - d(x_j, S_i)^2] \\ &= d(x_j, S)^2 - \frac{1}{m} \sum_{i=1}^m d(x_j, S_i)^2 \\ &= d(x_j, S)^2 - \frac{1}{m} d(0, \Phi(x_j))^2 \\ &\leq \left(1 - \frac{1}{m\bar{\gamma}^2}\right) d(x_j, S)^2, \end{aligned}$$

using the definition of metric subregularity.  $\square$

For a moment, let  $m = 2$  and consider the sequence of iterates  $\{x_j\}_{j \geq 0}$  generated by the randomized iterated projection algorithm. By idempotency of the projection operator, there's no benefit to projecting onto the same set in two consecutive iterations, so the subsequence consisting of different iterates corresponds exactly to that of the non-randomized iterated projection algorithm. In particular, if  $x_j \in S_1$ , then

$$d(P_{S_2}(x_j), S_1 \cap S_2)^2 \leq d(x_j, S_1 \cap S_2)^2 - d(x_j, S_2)^2 = d(x_j, S_1 \cap S_2)^2 - [d(x_j, S_2)^2 + d(x_j, S_1)^2]$$

since  $d(x_j, S_1) = 0$ . This gives us the following corollary, which also follows through more standard deterministic arguments.



**Corollary 4.4.3** *If  $\Phi$ , as defined by Equation 2.3.26 for  $m = 2$ , is metrically subregular at  $\bar{x}$  for 0 and  $\bar{\gamma} > \text{Subreg } \Phi(\bar{x}|0)$ , then for  $x_0$  sufficiently close to  $\bar{x}$ , the sequence generated by the 2-set iterated projection algorithm is linearly convergent, satisfying*

$$d(x_{j+1}, S_1 \cap S_2)^2 \leq \left(1 - \frac{1}{\bar{\gamma}^2}\right) d(x_j, S_1 \cap S_2)^2.$$

Note that this is very similar to a result in [7, Thm. 3.12] which shows linear convergence under an assumption of bounded linear regularity with a similar convergence rate.

In a similar theme to Corollary 4.4.3, consider the following refined version of the  $m$ -set randomized algorithm. Suppose  $x_0 \in S_1$  and  $i_0 = 1$ . Then for  $j = 0, 1, 2, \dots$ , let  $i_{j+1}$  be chosen uniformly at random from  $\{1, \dots, m\} \setminus \{i_j\}$  and  $x_{j+1} = P_{S_{i_{j+1}}}(x_j)$ . Using an identical type of analysis, we obtain the following similar result.

**Corollary 4.4.4** *If  $\Phi$ , as defined in Equation 2.3.26, is metrically subregular at  $\bar{x}$  for 0 and  $\bar{\gamma} > \text{Subreg } \Phi(\bar{x}|0)$ , then for  $x_0$  sufficiently close to  $\bar{x}$ , the sequence generated by the refined  $m$ -set randomized iterated projection algorithm is linearly convergent in expectation, satisfying*

$$\mathbf{E}[d(x_{j+1}, \cap_i S_i)^2 \mid x_j, i_{j-1}] \leq \left(1 - \frac{1}{(m-1)\bar{\gamma}^2}\right) d(x_j, \cap_i S_i)^2.$$

A simple but effective product space formulation by Pierra in [89] has the benefit of reducing the problem of finding a point in the intersection of finitely many sets to the problem of finding a point in the intersection of 2 sets. Using the notation above, we consider the closed set in the product space given by

$$T = S_1 \times S_2 \times \dots \times S_m$$

and the subspace

$$L = \{Ax : x \in \mathbb{E}\}$$

where the linear mapping  $A : \mathbb{E} \rightarrow \mathbb{E}^m$  is defined by  $Ax = (x, x, \dots, x)$ . Again, notice that  $\bar{x} \in \cap_i S_i \Leftrightarrow (\bar{x}, \dots, \bar{x}) \in T \cap L$ . One interesting aspect of this formulation is that projections in the product space  $\mathbb{E}^m$  relate back to projections in the original space  $\mathbb{E}$  by

$$\begin{aligned} (z_1, \dots, z_m) \in P_T(Ax) &\Leftrightarrow z_i \in P_{S_i}(x) \quad (i = 1, 2, \dots, m) \\ (P_L(z_1, \dots, z_m))_i &= \frac{1}{m}(z_1 + z_2 + \dots + z_m) \quad (i = 1, \dots, m) \end{aligned}$$

This formulation provides a nice analytical framework: we can use the above equivalence of projections to consider the *method of averaged projections* directly, defined as follows.

**Algorithm 4.4.5** Let  $S_1, \dots, S_m \subseteq \mathbb{E}$  be nonempty, closed, convex sets. Let  $x_0$  be an initial point. For  $j = 0, 1, 2, \dots$ , let

$$x_{j+1} = \frac{1}{m} \sum_{i=1}^m P_{S_i}(x_j).$$

Simply put, at each iteration, the algorithm projects the current iterate onto each set individually and takes the average of those projections as the next iterate. In the product space formulation, this is equivalent to  $x_{j+1} = P_L(P_T(x_j))$ . Expanding on the work of Pierra in [89], additional convergence theory for this algorithm has been examined by Bauschke and Borwein in [7]. Under appropriate regularity conditions, the general idea is that convergence of the iterated projection algorithm for two sets implies convergence of the averaged projection algorithm for  $m$  sets. In a similar sense, we prove the following result in terms of randomized projections.

**Theorem 4.4.6** *Suppose the assumptions of Theorem 4.4.2 hold. Then the conclusions of Theorem 4.4.2 hold for Algorithm 4.4.5 as well.*

**Proof** Let  $x_j$  be the current iterate,  $x_{j+1}^{AP}$  be the new iterate in the method of averaged projections and  $x_{j+1}^{RP}$  be the new iterate in the method of uniformly randomized projections. Then note that

$$x_{j+1}^{AP} = \frac{1}{m} \sum_{i=1}^m P_{S_i}(x_j) = \mathbf{E}[x_{j+1}^{RP} | x_j].$$

By convexity of the  $S_i$ 's, hence of the corresponding squared distance functions, it follows from Proposition 2.3.16 that

$$d(x_{j+1}^{AP}, S)^2 = d(\mathbf{E}[x_{j+1}^{RP} | x_j], S)^2 \leq \mathbf{E}[d(x_{j+1}^{RP}, S)^2 | x_j] \leq (1 - \frac{1}{m\bar{\gamma}^2})d(x_j, S)^2.$$

□

Hence, the method of averaged projections converges at least as quickly as the method of uniformly random projections. In particular, under the assumptions of Theorem 4.4.2, the method of averaged projections converges with rate no larger than  $1 - \frac{1}{m\bar{\gamma}^2}$ .

## 4.5 Reflection Methods

In this section, we will examine a generalization of the results on randomized projections in Section 4.4. Again, let  $S_i$  and  $P_{S_i}$ , for  $i = 1, \dots, m$ , be closed convex sets and the associated projection operators. Define the **reflection operator** for  $S_i$  by  $R_{S_i} := 2P_{S_i} - I$ , where  $I$  is the identity mapping. Note that

the  $R_{S_i}(x)$  is the unique point that maps  $x$  in the direction of  $P_{S_i}(x)$  satisfying  $\|x - P_{S_i}(x)\| = \|R_{S_i}(x) - P_{S_i}(x)\|$ .

Projection- and reflection-based algorithms share an interesting historical connection. While Kaczmarz was first proposing an iterated projection algorithm for solving linear systems in [63], Cimmino was first studying averaged reflection algorithms—similar in concept to the averaged projection algorithm examined in Algorithm 4.4.5—in [24]. A survey of Cimmino’s work can be found in [16].

To provide a mathematical connection, an iterated projection scheme, like that of the previous section, could be generically defined by taking an initial point  $x_0$  and iteratively finding the new point given by  $x_{j+1} = P_{S_i}(x_j)$  for some  $i \in \{1, \dots, m\}$ . Expressing this in terms of the reflection operator, the iteration can be equivalently described as  $x_{j+1} = \frac{1}{2}(I + R_{S_i})(x)$  for some  $i \in \{1, \dots, m\}$ .

Since the reflection operator is non-expansive and the composition of non-expansive operators remains non-expansive by Propositions 2.3.4 and 2.3.5, this suggests one possible generalization of an iterated projection algorithm can be obtained by replacing the reflection operator in the above description of the algorithm with a composition of reflections, leading to an iteration of the form

$$x_{j+1} = \frac{1}{2}(I + R_{S_{i_1}} R_{S_{i_2}} \dots R_{S_{i_k}})(x_j),$$

where  $i_1, \dots, i_k \in \{1, \dots, m\}$ . Note that this mapping is firmly non-expansive by Proposition 2.3.4. In the case of two sets, algorithms of this type were considered in [12], [74] and [13], among others. We will proceed in the same theme as in Sections 4.3 and 4.4; the remainder of this section is dedicated to analyzing the convergence of algorithms of this form, first in a simplified form for linear constraints and then for general convex feasibility problems under appropriate

regularity assumptions, namely metric regularity.

### 4.5.1 Linear Constraints

Consider the problem of solving a linear equality system  $Ax = b$ . For notational simplicity, let  $P_i$  and  $R_i$  denote the projection and reflection operators with respect to hyperplane  $i$ . Then, in a similar theme as Algorithm 4.3.1, consider the following randomized variant of the averaged reflections algorithm.

**Algorithm 4.5.1** *Consider a system of linear equalities  $Ax = b$  with  $A \neq 0$  and let  $x_0$  be an arbitrary initial point. For  $j = 0, 1, 2, \dots$ , compute*

$$x_{j+1} = \frac{1}{2} [x_j + R_i(R_k(x_j))]$$

*where, at each iteration  $j$ , the indices  $i$  and  $k$  are chosen independently at random from  $\{1, 2, \dots, m\}$  according to the distribution*

$$P\{i = t\} = P\{k = t\} = \frac{\|a_t\|^2}{\|A\|_F^2} \quad i, k = 1, \dots, m.$$

**Theorem 4.5.2** *Suppose the system  $Ax = b$  is feasible, with  $\text{rank}(A) > 1$  and let  $S = \{x : Ax = b\}$ . Then the sequence of iterates generated by Algorithm 4.5.1 is linearly convergent in expectation, satisfying*

$$\mathbf{E}[d(x_{j+1}, S)^2 \mid x_j] \leq \left(1 - \frac{2\sigma(A)^2}{\|A\|_F^2} + \frac{2\sigma(A)^4}{\|A\|_F^4}\right) d(x_j, S)^2.$$

*In particular, if  $A$  has full column rank, then*

$$\mathbf{E}[\|x_{j+1} - x^*\|^2 \mid x_j] \leq \left(1 - \frac{2}{\kappa(A)^2} + \frac{2}{\kappa(A)^4}\right) \|x_j - x^*\|^2,$$

*where  $x^*$  is the unique solution satisfying  $Ax^* = b$ .*

**Proof** Suppose the random operator chosen at iteration  $j$  is  $\frac{1}{2}[x_j + R_i(R_k(x_j))]$ . Taking any  $x^* \in S$  and recalling that this operator is firmly non-expansive, it follows that

$$\begin{aligned}\|x_{j+1} - x^*\|^2 &\leq \|x_j - x^*\|^2 - \left\| \frac{1}{2}(x_j - R_i(R_k(x_j))) \right\|^2 \\ &= \|x_j - x^*\|^2 - \left\| \frac{1}{2}(x_j - 2P_i(R_k(x_j)) + (2P_k(x_j) - x_j)) \right\|^2 \\ &= \|x_j - x^*\|^2 - \|P_k(x_j) - P_i(R_k(x_j))\|^2,\end{aligned}$$

from which we obtain

$$d(x_{j+1}, S)^2 \leq d(x_j, S)^2 - \|P_k(x_j) - P_i(R_k(x_j))\|^2. \quad (4.5.3)$$

Next, noting that

$$P_k(x_j) = x_j + \frac{(b_k - a_k^T x_j)}{\|a_k\|^2} a_k$$

and

$$R_k(x_j) = x_j + 2 \frac{(b_k - a_k^T x_j)}{\|a_k\|^2} a_k,$$

it follows that

$$P_k(x_j) - P_i(R_k(x_j)) = \frac{(a_k^T x_j - b_k)}{\|a_k\|^2} a_k + \frac{(a_i^T x_j - b_i)}{\|a_i\|^2} a_i - 2 \frac{(a_k^T x_j - b_k) a_i^T a_k}{\|a_k\|^2 \|a_i\|^2} a_i$$

and, therefore,

$$\|P_k(x_j) - P_i(R_k(x_j))\|^2 = \frac{(a_k^T x_j - b_k)^2}{\|a_k\|^2} + \frac{(a_i^T x_j - b_i)^2}{\|a_i\|^2} - 2 \frac{(a_i^T x_j - b_i)(a_k^T x_j - b_k)(a_i^T a_k)}{\|a_i\|^2 \|a_k\|^2}.$$

Taking the expectation with respect to the specified probability distribution, it follows that

$$\begin{aligned}E[\|P_k(x_j) - P_i(R_k(x_j))\|^2 \mid x_j] &= \frac{2}{\|A\|_F^2} \left[ \|Ax_j - b\|^2 - \frac{1}{\|A\|_F^2} \|A^T(Ax_j - b)\|^2 \right] \\ &= \frac{2}{\|A\|_F^2} (x_j - P_S(x_j))^T \left[ A^T A - \frac{1}{\|A\|_F^2} (A^T A)^2 \right] (x_j - P_S(x_j)) \\ &\geq \frac{2}{\|A\|_F^2} \underline{\sigma}(A^T A - \frac{1}{\|A\|_F^2} (A^T A)^2) d(x_j, S)^2 \\ &= \frac{2}{\|A\|_F^2} \left( \underline{\sigma}(A)^2 - \frac{1}{\|A\|_F^2} \underline{\sigma}(A)^4 \right) d(x_j, S)^2.\end{aligned}$$

Combining this with Inequality 4.5.3 provides the first result. The second result follows from the definition of  $\kappa(A)$  when  $A$  is full column rank.  $\square$

Note that the assumption that  $\text{rank}(A) > 1$  is necessary and sufficient for the convergence rate to be strictly less than 1, ensuring convergence.

It should be noted that the proven convergence rate isn't theoretically better (in expectation) than what would be obtained by performing two iterations of the randomized projection algorithm. In Figure 4.3, we compare a variety of algorithms on a randomly generated  $500 \times 200$  linear equality system. In particular, we compare the randomized projections Algorithm 4.3.1, the averaged randomized reflections Algorithm 4.5.1 and a variant of the averaged randomized reflections algorithm that composes three reflections at each iteration instead of two.

We also consider algorithms of this type for inequality systems. In Figure 4.4, we compare the same algorithms as above (where the projection and reflection operators are now with respect to the halfspaces instead of hyperplanes) for a randomly generated  $500 \times 50$  linear inequality system  $Ax \leq b$  where  $b$  is such that the feasible region has non-empty interior. Additionally, we compare an implementation where the hyperplanes are chosen with probability proportional to the squared row norms (solid lines in Figure 4.4 with an implementation incorporating a uniform distribution (circles in Figure 4.4)). For this example, we can see that the weighted implementation of the averaged reflection algorithms outperforms the uniformly randomized variants.

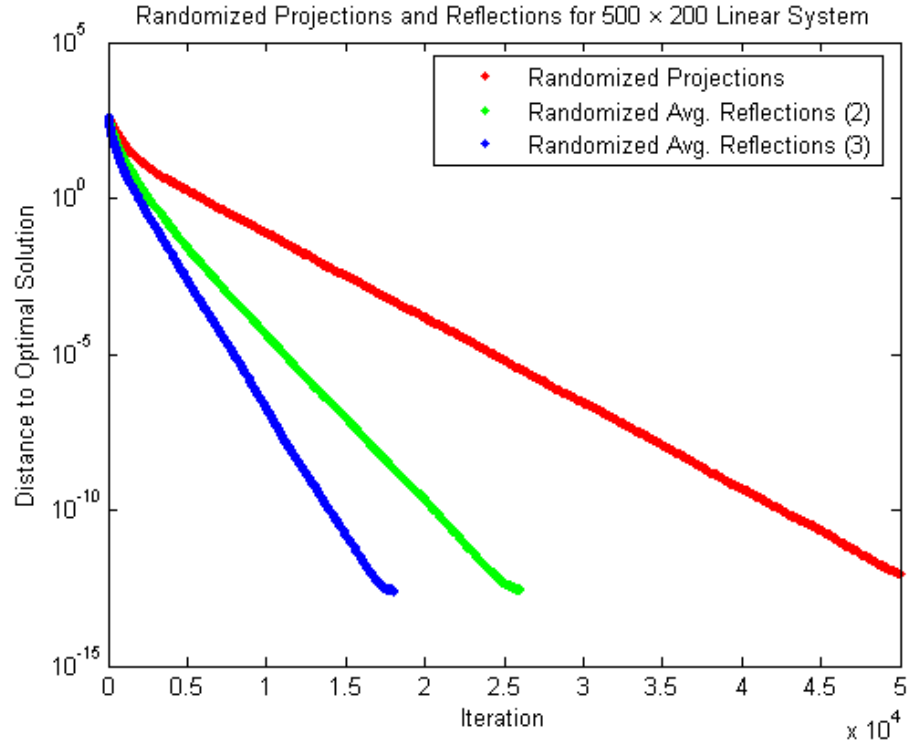


Figure 4.3: Randomized Reflections of Equality Systems

## 4.5.2 Convex Constraints

Let  $S_1, \dots, S_m$  be closed convex sets and return to the problem of finding  $x \in \cap_i S_i$ . In continuing with the theme of averaged reflection algorithms, consider the following algorithm.

**Algorithm 4.5.4** Consider an arbitrary initial point  $x_0$ . For  $j = 0, 1, 2, \dots$ , let

$$x_{j+1} = \frac{1}{2} \left( I + R_{S_{i_1}} R_{S_{i_2}} \dots R_{S_{i_k}} \right) (x_j)$$

where  $i_k$  is chosen uniformly at random from  $\{1, \dots, m\}$  and  $i_1, \dots, i_{k-1}$  are mutually distinct indices, different from  $i_k$ , chosen arbitrarily.



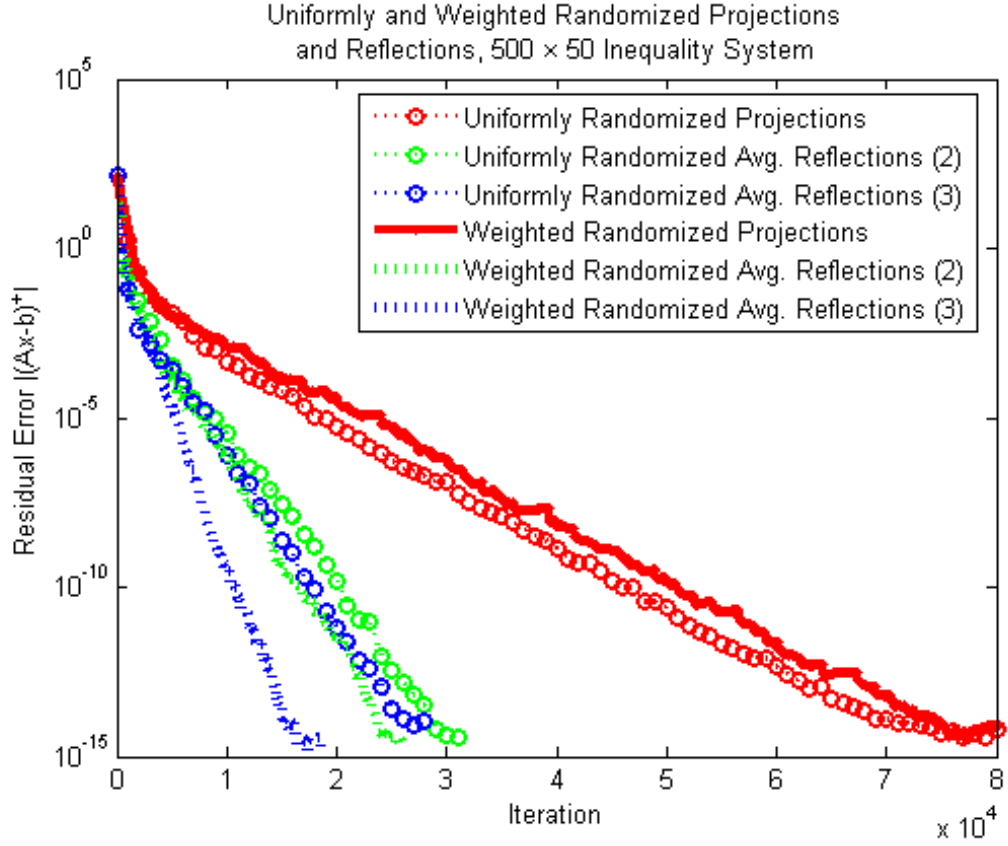


Figure 4.4: Randomized Reflections for Inequality Systems

Under appropriate regularity assumptions, we obtain the following local convergence result.

**Theorem 4.5.5** *Suppose  $\Phi(x) = [S_1 - x, \dots, S_m - x]^T$  is metrically regular at  $\bar{x}$  for 0 and let  $\bar{\gamma} > \text{Reg } \Phi(\bar{x}|0)$ . Further, suppose the initial point,  $x_0$ , is sufficiently close to  $\bar{x}$ . Then the sequence of iterates generated by Algorithm 4.5.4 is linearly convergent in expectation, satisfying*

$$\mathbf{E}[d(x_{j+1}, S)^2 \mid x_j] \leq \left(1 - \frac{1}{m\bar{\gamma}^4}\right) d(x_j, S)^2.$$

**Proof** Suppose that  $i_k$  is the random index chosen at iteration  $j$  and, for notational simplicity, define the operator  $T$  such that  $T = \frac{1}{2}(I + R_{S_{i_1}} R_{S_{i_2}} \dots R_{S_{i_k}})$ . Since

$T$  is firmly non-expansive for any choice of  $i_1, \dots, i_k$ , it follows from Inequality 2.3.2 that, for any  $x^* \in \cap_i S_{i_r}$ ,

$$d(T(x_j), S)^2 \leq \|T(x_j) - x^*\|^2 \leq \|x_j - x^*\|^2 - \|(I - T)(x_j)\|^2$$

and, choosing  $x^* = P_S(x_j)$ , we obtain

$$d(T(x_j), S)^2 \leq d(x_j, S)^2 - \|(I - T)(x_j)\|^2. \quad (4.5.6)$$

For some notation, define  $R^{k+1}(x_j) = x_j$  and let

$$R^p(x_j) = R_{S_{i_p}} R_{S_{i_{p+1}}} \dots R_{S_{i_k}}(x_j) \text{ for } p = 1, \dots, k.$$

Next, observe that

$$\begin{aligned} x_j - T(x_j) &= \frac{1}{2}[R^{k+1}(x_j) - R^1(x_j)] \\ &= \frac{1}{2} \sum_{p=1}^k [R^{p+1}(x_j) - R^p(x_j)] \\ &= \sum_{p=1}^k [R^{p+1}(x_j) - P_{S_{i_p}}(R^{p+1}(x_j))] \\ &= \sum_{p=1}^k (I - P_{S_{i_p}})(R^{p+1}(x_j)). \end{aligned}$$

By the non-expansivity of the reflection operators, it follows that if  $x_0$  is sufficiently close to  $\bar{x}$  so that Inequality 2.3.18 holds with constant  $\bar{\gamma}$ , then  $x_j$  and  $R^p(x_j)$  are as well for any  $p = 1, \dots, k$  and any  $j \geq 0$ . Further, recall that for any  $p$ ,  $(I - P_{S_{i_p}})(R^{p+1}(x_j)) \in N_{S_{i_p}}(P_{S_{i_p}}(R^{p+1}(x_j)))$ . Therefore, we can apply Corollary 2.3.30 to see that

$$\|x_j - T(x_j)\|^2 \geq \frac{1}{\bar{\gamma}^2} \sum_{p=1}^k \|(I - P_{S_{i_p}})(R^{p+1}(x_j))\|^2 = \frac{1}{\bar{\gamma}^2} \sum_{p=1}^k d(R^{p+1}(x_j), S_{i_p})^2.$$

Combining this with Inequality 4.5.6, we obtain

$$d(T(x_j), S)^2 \leq d(x_j, S)^2 - \frac{1}{\bar{\gamma}^2} \sum_{p=1}^k d(R^{p+1}(x_j), S_{i_p})^2 \quad (4.5.7)$$

$$\leq d(x_j, S)^2 - \frac{1}{\bar{\gamma}^2} d(x_j, S_{i_k})^2. \quad (4.5.8)$$

According to the probability distribution specified in the algorithm,  $i_k$  is uniformly distributed on  $\{1, \dots, m\}$ . Taking the expected value with respect to this distribution, we obtain

$$\begin{aligned} \mathbf{E}[d(x_{j+1}, S)^2 \mid x_j] &\leq d(x_j, S)^2 - \frac{1}{m\bar{\gamma}^2} \sum_i d(x_j, S_i)^2 \\ &= d(x_j, S)^2 - \frac{1}{m\bar{\gamma}^2} d(0, \Phi(x_j))^2 \\ &\leq \left(1 - \frac{1}{m\bar{\gamma}^4}\right) d(x_j, S)^2, \end{aligned}$$

providing the desired result.  $\square$

## 4.6 Concluding Remarks

The relationship between the speed of certain iterative algorithms, error bounds and the distance to ill-posedness has been well-studied within the optimization community. In this chapter, we expanded upon this relationship through the framework of randomized algorithms. As a motivating example, for the problem of solving a positive definite linear system,  $Ax = b$ , a certain randomized coordinate descent algorithm was shown to be linearly convergent in expectation with rate  $1 - \frac{1}{\|A^{-1}\|_2 \text{tr}(A)}$ , which can further be bounded in terms of the natural, algebraic conditioning measures by  $1 - \frac{1}{\sqrt{n\kappa(A)}} \leq 1 - \frac{1}{n \kappa(A)}$ .

Expanding to the case of full-rank linear systems  $Ax = b$ , a coordinate descent algorithm was also shown to be linearly convergent in expectation with rate  $1 - \frac{1}{\kappa(A)^2}$ , matching a result by Strohmer and Vershynin in [109] for a randomized projections algorithm. Further, a randomized variant of an averaged reflections algorithm, requiring the computation of two projections per iteration,

was shown to be linearly convergent with rate  $1 - \frac{2}{\kappa(A)^2} + \frac{2}{\kappa(A)^4}$ . By generalizing the randomized projections scheme to mixed linear equality and inequality systems, linearly convergence in expectation is again obtained with rate  $1 - \frac{1}{\|A\|_F^2 L^2}$ , where  $L$  is Hoffman's error bound for the system as originally investigated in [58].

Randomized projection algorithms were then considered for solving convex feasibility problems. In particular, it was shown that a randomized projections scheme is linearly convergent in expectation with rate  $1 - \frac{1}{m\bar{\gamma}^2}$  and, through a slight refinement of the algorithm, the constant  $m$  can be replaced with  $m - 1$ , where  $\bar{\gamma}$  is the constant associated with the local error bound induced by the metric subregularity of a related set-valued mapping. Further, if that mapping is in fact metrically regular, it was further demonstrated that an averaged reflections scheme is also linearly convergent in expectation with rate  $1 - \frac{1}{m\bar{\gamma}^4}$ , where  $\bar{\gamma}$  is now the constant associated with the error bound induced by metric regularity.

## CHAPTER 5

### RANDOMIZED PROXIMAL POINT METHODS

#### 5.1 Introduction

In Section 4.4, we examined how certain regularity assumptions—specifically, metric subregularity of a specific set-valued mapping—can be used to demonstrate a convergence rate for a randomized projections algorithm. In this chapter, we will proceed by examining how to generalize some of those results. We will begin with some definitions.

**Definition 5.1.1** *A set-valued mapping  $T : \mathbb{E} \rightrightarrows \mathbb{E}$  is **monotone** if it satisfies*

$$\langle x_1 - x_0, y_1 - y_0 \rangle \geq 0 \text{ for all } x_0, x_1 \in \mathbb{E}, y_0 \in T(x_0), y_1 \in T(x_1). \quad (5.1.2)$$

**Definition 5.1.3** *A monotone operator,  $T$ , is called **maximal monotone** if there does not exist a monotone operator  $T' \neq T$  such that  $\text{gph } T \subseteq \text{gph } T'$ .*

Monotonicity can be thought of as a generalization of positive semi-definiteness to the class of set-valued mappings. For example, if  $f(x) = \frac{1}{2}x^T Ax + b^T x$  is a convex quadratic function—implying that  $A$  is positive semi-definite—then it follows from the definition that the gradient mapping  $\nabla f(x) = Ax + b$  is a monotone operator.

The next result by Rockafellar generalizes this observation regarding monotonicity from quadratic functions to a broad class of convex functions; first, however, we will provide some necessary definitions. A convex function

$f : \mathbb{E} \rightarrow [-\infty, \infty]$  is **lower semi-continuous at**  $\bar{x}$  if  $\liminf_{x \rightarrow \bar{x}} f(x) \geq f(\bar{x})$  and **proper** if  $\text{dom} f \neq \emptyset$  and  $f(x) \neq -\infty$  for all  $x \in \mathbb{E}$ .

**Theorem 5.1.4 ([101])** *If  $f$  is a proper, convex function and lower semi-continuous everywhere, then the subdifferential mapping  $\partial f$  is a maximal monotone operator.*

In this chapter, we will focus on the two monotone inclusion problems

$$\text{Find } x \in \mathbb{E} \text{ such that } 0 \in T(x) \quad (5.1.5)$$

and

$$\text{Find } x \in \mathbb{E} \text{ such that } 0 \in \cap_{i \in I} T_i(x), \quad (5.1.6)$$

where  $I$  is some index set and  $T, T_i : \mathbb{E} \rightrightarrows \mathbb{E}, i \in I$  are maximal monotone operators.

**Example 5.1.7** *Let  $S_1, \dots, S_m$  be closed, convex sets. By Example 2.3.15 and Theorem 5.1.4, each normal cone mapping  $N_{S_i}(\cdot) = \partial \iota_{S_i}(\cdot)$  is a maximal monotone operator. Therefore, solving the convex feasibility problem of finding  $x \in \cap_i S_i$  is equivalent to solving Problem 5.1.6 for the mappings  $T_i = N_{S_i}$ .*

For  $\lambda > 0$ , the mappings  $J_{\lambda T} := (I + \lambda T)^{-1}$  are the **resolvents** of  $T$ . It is easily seen that if  $T$  is monotone, then each resolvent is single-valued. Further, in many applications, solving  $b \in (I + \lambda T)(x)$  is easier than solving  $b \in T(x)$  for a given vector  $b$ . One proposed method for solving Problem 5.1.5 is the **proximal point algorithm**, considered originally in [80] and more thoroughly explored by [102], given by

$$x_{k+1} = J_{\lambda T}(x_k) \text{ for } k = 0, 1, 2, \dots \quad (5.1.8)$$

**Example 5.1.9** *Let  $S$  be a closed, convex set. By Proposition 2.3.11, the projection mapping satisfies  $x - P_S(x) \in N_S(P_S(x))$ . Alternatively, this can be expressed as  $P_S(x) \in (I + N_S)^{-1}(x)$ . Since the normal cone mapping is a cone, implying that  $\lambda N_S(x) = N_S(x)$  for all  $\lambda > 0$ , it follows that every resolvent of the normal cone mapping is the projection operator.*

The class of (maximal) monotone operators is of particular interest due to their prominent role in convex analysis and applications to problems involving partial differential equations, convex minimization and solving variational inequalities. As the next example shows, the connection with convex minimization is immediate.

**Example 5.1.10** *Let  $f : \mathbb{E} \rightarrow [-\infty, \infty]$  be a proper, convex function that is lower semi-continuous everywhere. Then for a given vector  $x$ , it follows from Theorem 5.1.4 and the definition of the resolvent that*

$$J_{\lambda \partial f}(x) \in \operatorname{argmin}_{y \in \mathbb{E}} \left[ f(y) + \frac{1}{2\lambda} \|y - x\|^2 \right].$$

*Further, by strict convexity of the minimand, it follows that the above minimizer is unique.*

Motivated by the same principles as Section 4.4, the goal in this chapter is to examine how appropriate regularity assumptions on the operators  $T$  (or  $T_1, \dots, T_m$ , respectively) affect the speed of convergence of variants of the proximal point algorithm. To begin, we will cite some additional preliminary results on monotone operators and resolvents.

**Proposition 5.1.11 ([103])** *If  $T$  is maximal monotone, then  $T^{-1}$  is as well, in which case both  $T$  and  $T^{-1}$  are closed- and convex-valued.*

**Proposition 5.1.12 ([102],[38])** *The operator  $T$  is monotone if and only if the resolvent  $J_{\lambda T}$  is firmly non-expansive. Further,  $T$  is maximal monotone if and only if  $J_{\lambda T}$  is firmly non-expansive and  $\text{dom } J_{\lambda T} = \mathbb{E}$ .*

The next result, cited from [38] but resembling a result originally from [81], further details the above connection between a maximal monotone mapping and its resolvent.

**Proposition 5.1.13** *The mapping  $T \rightarrow (I + \lambda T)^{-1}$  is a bijection between the collection of maximal monotone operators and the collection of firmly non-expansive operators with full domain.*

## 5.2 Metric Subregularity and Linear Convergence

### 5.2.1 The Main Results

We now return to Problem 5.1.5, the problem of finding a zero of a maximal monotone operator. Variants of proximal point algorithms for solving this problem have been considered by a wide variety of authors, including [102], [78], [106], [88], [4] and others.

Many authors consider an algorithmic framework much more general than the one considered in this paper. Some of the most studied variants allow for a varying proximal parameter  $\lambda$ , allow approximate computation of the proximal iteration, allow over- or under-relaxation in the proximal step or incorporate an additional projective framework. These ideas have often proven worthwhile



both for designing computationally practical and efficient algorithms as well as for improving convergence analysis. However, in this paper, we will only consider algorithms in their “classical” form, assuming exact computation of the resolvent with a fixed proximal parameter. Our particular interest is in exploring how naturally occurring constants—for example, the modulus of subregularity of the mappings themselves and regularity conditions associated with the solution sets—govern the local rates of convergence and, further, how randomization as an analytical tool can emphasize this connection. To begin, consider the basic proximal point algorithm given by 5.1.8, where  $x_{k+1} = J_{\lambda T}(x_k)$ . Under an assumption of metric subregularity, we obtain the following initial result.

**Theorem 5.2.1** *Suppose  $T$  is maximal monotone and metrically subregular at  $\bar{x}$  for 0 with subregularity modulus  $\text{Subreg } T(\bar{x}|0)$ . Let  $\bar{\gamma} > \text{Subreg } T(\bar{x}|0)$  and suppose  $x_0$  is sufficiently near  $\bar{x}$ . Then the sequence of iterates generated by Algorithm 5.1.8 is linearly convergent to  $T^{-1}(0)$ , the zero-set of  $T$ , satisfying*

$$d(x_{k+1}, T^{-1}(0))^2 \leq \frac{\bar{\gamma}^2}{\lambda^2 + \bar{\gamma}^2} d(x_k, T^{-1}(0))^2.$$

**Proof** Let  $\hat{x} \in T^{-1}(0)$  and note that  $J_{\lambda T}(\hat{x}) = \hat{x}$ . Since the resolvent of a monotone operator is firmly non-expansive, it follows that, for any  $x$ ,

$$\|J_{\lambda T}(x) - J_{\lambda T}(\hat{x})\|^2 \leq \|x - \hat{x}\|^2 - \|(I - J_{\lambda T})(x) - (I - J_{\lambda T})(\hat{x})\|^2,$$

implying that

$$\|J_{\lambda T}(x) - \hat{x}\|^2 \leq \|x - \hat{x}\|^2 - \|x - J_{\lambda T}(x)\|^2. \quad (5.2.2)$$

However, by definition of  $J_{\lambda T}$ ,

$$x - J_{\lambda T}(x) \in \lambda T(J_{\lambda T}(x)).$$

In particular,

$$\|x - J_{\lambda T}(x)\| \geq \lambda \min\{\|z\| : z \in T(J_{\lambda T}(x))\} = \lambda d(0, T(J_{\lambda T}(x))). \quad (5.2.3)$$

Now, note that since the resolvents and projection operators are firmly non-expansive, if  $x_0$  is sufficiently close to  $\bar{x}$  such that Inequality 2.3.22 holds with constant  $\bar{\gamma}$ , then  $x_j$  and  $P_{T^{-1}(0)}(x_j)$  are as well for each  $j \geq 0$ . Therefore, it follows that

$$\begin{aligned} d(x_{k+1}, T^{-1}(0))^2 &\leq \|x_{k+1} - P_{T^{-1}(0)}(x_k)\|^2 \\ &\leq \|x_k - P_{T^{-1}(0)}(x_k)\|^2 - \|x_k - J_{\lambda T}(x_k)\|^2 \quad (\text{Inequality 5.2.2}) \\ &\leq d(x_k, T^{-1}(0))^2 - \lambda^2 d(0, T(J_{\lambda T}(x_k)))^2 \quad (\text{Inequality 5.2.3}) \\ &\leq d(x_k, T^{-1}(0))^2 - \frac{\lambda^2}{\bar{\gamma}^2} d(J_{\lambda T}(x_k), T^{-1}(0))^2 \quad (\text{Inequality 2.3.22}) \\ &= d(x_k, T^{-1}(0))^2 - \frac{\lambda^2}{\bar{\gamma}^2} d(x_{k+1}, T^{-1}(0))^2. \end{aligned}$$

This implies that

$$(1 + \frac{\lambda^2}{\bar{\gamma}^2}) d(x_{k+1}, T^{-1}(0))^2 \leq d(x_k, T^{-1}(0))^2,$$

from which the result follows.  $\square$

Further observe that by considering a sequence  $\{\lambda_k\}$  such that  $\lambda_k \rightarrow \infty$  instead of a fixed  $\lambda$  in the above algorithm, we obtain superlinear convergence.

Our primary interest in Theorem 5.2.1 is as a tool in proving the following result, Theorem 5.2.5. However, we note that Theorem 5.2.1 is similar to some previously known results. For example, linear convergence was shown in [102] and [106], under a framework that permitted error in evaluating the resolvent, with a slightly stronger regularity assumption. In particular, as a limiting case

with no such error in evaluating the resolvent, an identical convergence rate was obtained in [102]. The result by Solodov and Svaiter in [106], however, corresponds to a hybrid proximal-projection algorithm.

We wish to generalize this result to Problem 5.1.6, that of finding a common zero among a finite set of maximal monotone operators,  $T_1, \dots, T_m$ . Variants of proximal point algorithms for this problem have been considered by a variety of authors, including [65], [69], [106], [26], [56], among others. We will consider the following randomized variant of a proximal point algorithm: for  $k = 0, 1, 2, \dots$ ,

$$x_{k+1} = J_{\lambda T_i}(x_k) \quad \text{with probability } \frac{1}{m}, \quad i = 1, \dots, m. \quad (5.2.4)$$

Based on the discussion in Section 5.1, when each  $T_i$  is the normal cone mapping to a closed and convex set, this is exactly the randomized projections algorithm of Theorem 4.4.2; the connection between projections and proximal point methods will be discussed further in Subsection 5.2.2. In the general form, we obtain the following result.

**Theorem 5.2.5** *Suppose the following assumptions hold:*

1. *The maximal monotone operators  $\{T_i : i = 1, \dots, m\}$ , are metrically subregular at  $\bar{x} \in \cap_j T_j^{-1}(0)$  for 0 with respective moduli  $\text{Subreg } T_i(\bar{x}|0)$ .*
2. *The mapping  $\Phi(x) = [T_1^{-1}(0) - x, \dots, T_m^{-1}(0) - x]^T$  is metrically subregular at  $\bar{x}$  for 0 with modulus  $\text{Subreg } \Phi(\bar{x}|0)$ .*
3.  *$\bar{\gamma} > \max\{\text{Subreg } T_i(\bar{x}|0) : i = 1, \dots, m\}$  and  $\bar{\kappa} > \text{Subreg } \Phi(\bar{x}|0)$ .*

*Then for  $x_0$  sufficiently close to  $\bar{x}$ , the sequence of iterates generated by Algorithm 5.2.4 is linearly convergent in expectation to the common zero set,  $\cap_j T_j^{-1}(0)$ , satisfying*

$$\mathbf{E}[d(x_{k+1}, \cap_j T_j^{-1}(0))^2 \mid x_k] \leq \left(1 - \frac{1}{m\bar{\kappa}^2} \left[1 - \left(\frac{\bar{\gamma}^2}{\lambda^2 + \bar{\gamma}^2}\right)^{\frac{1}{2}}\right]^2\right) d(x_k, \cap_j T_j^{-1}(0))^2.$$

**Proof** If  $x_0$  is sufficiently close to  $\bar{x}$  so that Inequality 2.3.22 holds with constant  $\bar{\gamma}$ , it follows from the firm non-expansivity of the resolvents and the projection operator that each iterate  $x_k$  and the projection of each iterate onto the common zero set,  $P_{\cap_j T_j^{-1}(0)}(x_k)$ , are sufficiently close to  $\bar{x}$  as well.

Suppose that at iteration  $k$ , the resolvent  $J_{\lambda T_i}$  is chosen by the algorithm. Then it follows that

$$\begin{aligned}
d(J_{\lambda T_i}(x_k), \cap_j T_j^{-1}(0))^2 &= \|J_{\lambda T_i}(x_k) - P_{\cap_j T_j^{-1}(0)}(J_{\lambda T_i}(x_k))\|^2 \\
&\leq \|J_{\lambda T_i}(x_k) - P_{\cap_j T_j^{-1}(0)}(x_k)\|^2 \\
&\leq d(x_k, \cap_j T_j^{-1}(0))^2 - \|x_k - J_{\lambda T_i}(x_k)\|^2 \\
&= d(x_k, \cap_j T_j^{-1}(0))^2 - \left\| \left[ x_k - P_{T_i^{-1}(0)}(x_k) \right] + \left[ P_{T_i^{-1}(0)}(x_k) - J_{\lambda T_i}(x_k) \right] \right\|^2 \\
&\leq d(x_k, \cap_j T_j^{-1}(0))^2 - d(x_k, T_i^{-1}(0))^2 - \|P_{T_i^{-1}(0)}(x_k) - J_{\lambda T_i}(x_k)\|^2 \\
&\quad - 2\langle x_k - P_{T_i^{-1}(0)}(x_k), P_{T_i^{-1}(0)}(x_k) - J_{\lambda T_i}(x_k) \rangle \\
&\leq d(x_k, \cap_j T_j^{-1}(0))^2 - d(x_k, T_i^{-1}(0))^2 - \|P_{T_i^{-1}(0)}(J_{\lambda T_i}(x_k)) - J_{\lambda T_i}(x_k)\|^2 \\
&\quad - 2\langle x_k - P_{T_i^{-1}(0)}(x_k), P_{T_i^{-1}(0)}(x_k) - J_{\lambda T_i}(x_k) \rangle.
\end{aligned}$$

Note that

$$\begin{aligned}
&-2\langle x_k - P_{T_i^{-1}(0)}(x_k), P_{T_i^{-1}(0)}(x_k) - J_{\lambda T_i}(x_k) \rangle \\
&= 2\langle x_k - P_{T_i^{-1}(0)}(x_k), [J_{\lambda T_i}(x_k) - P_{T_i^{-1}(0)}(J_{\lambda T_i}(x_k))] + [P_{T_i^{-1}(0)}(J_{\lambda T_i}(x_k)) - P_{T_i^{-1}(0)}(x_k)] \rangle \\
&\leq 2\langle x_k - P_{T_i^{-1}(0)}(x_k), J_{\lambda T_i}(x_k) - P_{T_i^{-1}(0)}(J_{\lambda T_i}(x_k)) \rangle \\
&\leq 2\|x_k - P_{T_i^{-1}(0)}(x_k)\| \|J_{\lambda T_i}(x_k) - P_{T_i^{-1}(0)}(J_{\lambda T_i}(x_k))\| \\
&= 2d(x_k, T_i^{-1}(0)) d(J_{\lambda T_i}(x_k), T_i^{-1}(0)).
\end{aligned}$$

The first inequality comes from the fact that  $x_k - P_{T_i^{-1}(0)}(x_k) \in N_{T_i^{-1}(0)}(P_{T_i^{-1}(0)}(x_k))$  so Inequality 2.3.9 can be applied from the definition of the normal cone. The second inequality is an application of the Cauchy-Schwartz inequality. The rest

follows from the definition of the projection operator. Putting this together, we obtain

$$\begin{aligned}
d(J_{\lambda T_i}(x_k), \cap_j T_j^{-1}(0))^2 &\leq d(x_k, \cap_j T_j^{-1}(0))^2 - d(x_k, T_i^{-1}(0))^2 - d(J_{\lambda T_i}(x_k), T_i^{-1}(0))^2 \\
&\quad + 2 d(x_k, T_i^{-1}(0)) d(J_{\lambda T_i}(x_k), T_i^{-1}(0)) \\
&= d(x_k, \cap_j T_j^{-1}(0))^2 - \left( d(x_k, T_i^{-1}(0)) - d(J_{\lambda T_i}(x_k), T_i^{-1}(0)) \right)^2.
\end{aligned}$$

Noting that  $d(x_k, T_i^{-1}(0)) - d(J_{\lambda T_i}(x_k), T_i^{-1}(0)) \geq 0$  always, it follows from an application of Theorem 5.2.1 that

$$d(J_{\lambda T_i}(x_k), \cap_j T_j^{-1}(0))^2 \leq d(x_k, \cap_j T_j^{-1}(0))^2 - \left[ 1 - \left( \frac{\bar{\gamma}^2}{\lambda^2 + \bar{\gamma}^2} \right)^{\frac{1}{2}} \right]^2 d(x_k, T_i^{-1}(0))^2.$$

Taking the expected value, we obtain

$$\begin{aligned}
\mathbf{E}[d(x_{k+1}, \cap_j T_j^{-1}(0))^2 \mid x_k] &\leq d(x_k, \cap_j T_j^{-1}(0))^2 - \frac{1}{m} \left[ 1 - \left( \frac{\bar{\gamma}^2}{\lambda^2 + \bar{\gamma}^2} \right)^{\frac{1}{2}} \right]^2 \sum_{i=1}^m d(x_k, T_i^{-1}(0))^2 \\
&= d(x_k, \cap_j T_j^{-1}(0))^2 - \frac{1}{m} \left[ 1 - \left( \frac{\bar{\gamma}^2}{\lambda^2 + \bar{\gamma}^2} \right)^{\frac{1}{2}} \right]^2 d(0, \Phi(x_k))^2 \\
&\leq \left( 1 - \frac{1}{m\bar{\kappa}^2} \left[ 1 - \left( \frac{\bar{\gamma}^2}{\lambda^2 + \bar{\gamma}^2} \right)^{\frac{1}{2}} \right]^2 \right) d(x_k, \cap_j T_j^{-1}(0))^2,
\end{aligned}$$

where the last inequality follows from the metric subregularity of the mapping  $\Phi(x) = [T_1^{-1}(0) - x, \dots, T_m^{-1}(0) - x]^T$ .  $\square$

One particularly simple way of de-randomizing Algorithm 5.2.4 is by considering averaged resolvents or, in the terminology of [69], the *barycentric proximal method*. Specifically, given maximal monotone operators  $T_i$ ,  $i = 1, \dots, m$  with respective resolvents  $J_{\lambda T_i}$ ,  $i = 1, \dots, m$ , consider the algorithm described such that, for  $k = 0, 1, 2, \dots$ ,

$$x_{k+1} = \frac{1}{m} \sum_{i=1}^m J_{\lambda T_i}(x_k) \tag{5.2.6}$$

and the associated fixed-point problem

$$\text{Find } x \in \mathbb{E} \text{ such that } x = \frac{1}{m} \sum_{i=1}^m J_{\lambda T_i}(x). \tag{5.2.7}$$

The following proposition, found in [69], provides the necessary connection.

**Proposition 5.2.8 ([69])** *If  $\bar{x} \in \cap_i T_i^{-1}(0)$ , then  $\bar{x}$  is a solution to Problem 5.2.7. Further, if  $\cap_i T_i^{-1}(0) \neq \emptyset$ , the fixed points of Problem 5.2.7 are common zero points of all the  $T_i$ 's.*

Again returning to the case where each operator  $T_i$  is the normal cone mapping for some closed, convex set, it follows that Algorithm 5.2.6 is simply the *averaged projections algorithm* studied by [89], [95], [7], [70], among others, as well as in Section 4.4. More generally, we can use the result of Theorem 5.2.5 to generalize the result on averaged projections found in Theorem 4.4.6 to the barycentric proximal method.

**Theorem 5.2.9** *Suppose the assumptions of Theorem 5.2.5 hold. Then the conclusions of Theorem 5.2.5 hold for Algorithm 5.2.6 as well.*

**Proof** Let  $x_k$  be the current iterate,  $x_{k+1}^{BP}$  be the new iterate in the barycentric proximal method, Algorithm 5.2.6, and let  $x_{k+1}^{RP}$  be the new iterate in the randomized proximal point method, Algorithm 5.2.4. First, note that since each set  $T_i^{-1}(0)$  is convex, the distance function  $d(\cdot, \cap_j T_j^{-1}(0))$  is as well, and

$$d(J_{\lambda T_i}(x_k), \cap_j T_j^{-1}(0)) \leq d(x_k, \cap_j T_j^{-1}(0)) \text{ for } i = 1, \dots, m,$$

from which it follows that

$$d(x_{k+1}^{BP}, \cap_j T_j^{-1}(0)) \leq d(x_k, \cap_j T_j^{-1}(0)).$$

Let  $\alpha = \left(1 - \frac{1}{mk^2} \left[1 - \left(\frac{\bar{\gamma}^2}{\lambda^2 + \bar{\gamma}^2}\right)^{\frac{1}{2}}\right]^2\right)$  and observe that the function  $d(\cdot, \cap_j T_j^{-1}(0))^2$  is also convex. Noting that

$$x_{k+1}^{BP} = \frac{1}{m} \sum_{j=1}^m J_{\lambda T_j}(x_k) = \mathbf{E}[x_{k+1}^{RP} \mid x_k],$$

it follows that

$$\begin{aligned}
d(x_{k+1}^{BP}, \cap_j T_j^{-1}(0))^2 &= d(\mathbf{E}[x_{k+1}^{RP} \mid x_k], \cap_j T_j^{-1}(0))^2 \\
&\leq \mathbf{E}[d(x_{k+1}^{RP}, \cap_j T_j^{-1}(0))^2 \mid x_k] \\
&\leq \alpha d(x_k, \cap_j T_j^{-1}(0))^2,
\end{aligned}$$

with the first inequality being an application of Jensen's Inequality and the second being an application of Theorem 5.2.5.  $\square$

Therefore, the barycentric proximal method converges at least as quickly as the randomized proximal point method.

## 5.2.2 Projection Algorithms: A Special Case

In Theorem 5.2.5, we demonstrated a linear convergence result for a randomized proximal point method for finding a common zero of multiple, maximal monotone operators. Further, we showed that the randomized projections algorithm of Theorem 4.4.2 is a special case of the randomized proximal point method. We will now show that the assumptions of these two results are equivalent in this special case, and therefore, that Theorem 5.2.5 implies Theorem 4.4.2.

By Examples 5.1.7 and 5.1.9, we know that the normal cone is a maximal monotone operator whose resolvent is the projection operator. First, we wish to show that for a closed, convex, non-empty set,  $S$ , the normal cone mapping is metrically subregular at all  $\bar{x} \in S$  for 0 with modulus zero. To see this, consider any  $x$  near  $\bar{x}$ . If  $x \in S$ , then  $d(x, (N_S)^{-1}(0)) = 0 = d(N_S(x), 0)$  trivially and if  $x \notin S$ ,

it follows that  $d(N_S(x), 0) = \infty$  since  $N_S(x) = \emptyset$  but  $d(x, (N_S)^{-1}(0)) = d(x, S) < \infty$ . Therefore, it follows that for all  $\gamma > 0$ ,  $d(x, (N_S)^{-1}(0)) \leq \gamma d(N_S(x), 0)$ , implying that the modulus of subregularity is zero.

This satisfies Assumption 1 of Theorem 5.2.5. Since  $(N_S)^{-1}(0) = S$ , Assumption 2 is satisfied by the metric subregularity of  $\Phi(x) = [S_1 - x, \dots, S_m - x]^T$  as stated in Theorem 4.4.2, as is the condition on  $\bar{\kappa}$  in Assumption 3. Choosing any  $\bar{\gamma} > 0$  and  $\lambda > 0$ , we see that all the assumptions do in fact hold.

This allows us to apply Theorem 5.2.5, the result on the randomized proximal point algorithm, to the randomized projection algorithm. However, since each resolvent is the same for any choice of  $\lambda$ , taking  $\lambda \rightarrow \infty$ , we in fact obtain a convergence rate identical to Theorem 4.4.2 on randomized projection algorithms.

However, for *other* applications, by taking the proximal parameter  $\lambda$  to be arbitrarily large, we still obtain a convergence rate asymptotically equal to that from the randomized projection algorithm, suggesting that there may be a deeper connection between proximal point methods and projection methods. In fact, as discussed in [106] among others, an exact-computation proximal point method can be thought of as a specific type of projection algorithm. The analysis begins with the following lemma.

**Proposition 5.2.10** *Suppose  $T$  is maximal monotone. Then*

$$\langle x - J_{\lambda T}(x), z - J_{\lambda T}(x) \rangle \leq 0 \quad \text{for all } z \in T^{-1}(0).$$

**Proof** Observe that for any  $z \in T^{-1}(0)$ ,

$$\langle x - J_{\lambda T}(x), z - J_{\lambda T}(x) \rangle = -\lambda \langle 0 - \frac{x - J_{\lambda T}(x)}{\lambda}, z - J_{\lambda T}(x) \rangle \leq 0,$$



from the fact that  $x - J_{\lambda T}(x) \in \lambda T(J_{\lambda T}(x))$ ,  $0 \in T(z)$  and the monotonicity of  $T$ .  $\square$

Observe that if  $x \notin T^{-1}(0)$  (implying  $x \neq J_{\lambda T}(x)$ ), the hyperplane

$$H = \{y : \langle x - J_{\lambda T}(x), y - J_{\lambda T}(x) \rangle = 0\}$$

strictly separates  $x$  from  $T^{-1}(0)$  by Lemma 5.2.10. Moreover, it is easy to verify that  $J_{\lambda T}(x) = P_H(x)$ . Hence, each proximal iteration is, in fact, a projection on to a particular hyperplane. Following along this line of thought, Proposition 5.2.10 further implies that if it were the case that  $J_{\lambda T}(x) \in T^{-1}(0)$ , then  $x - J_{\lambda T}(x) \in N_{T^{-1}(0)}(J_{\lambda T}(x))$ , implying that  $J_{\lambda T}(x) = P_{T^{-1}(0)}(x)$ . Given this connection between resolvents and projection operators, it would be noteworthy to see directly that as the proximal parameter gets arbitrarily large, the proximal point iteration becomes a projection onto  $T^{-1}(0)$ . In fact, this is shown in the following result, originally by Kido in [64].

**Theorem 5.2.11 ([64])** *Let  $T$  be a maximal monotone operator. Then*

$$\lim_{\lambda \rightarrow \infty} J_{\lambda T}(x) = P_{T^{-1}(0)}(x).$$

## CHAPTER 6

### OPEN QUESTIONS AND FUTURE RESEARCH

The following is a list of ideas to serve as a stimulus for future research. The items are loosely presented in increasing order of the interest we have in them.

1. There is a quantifiable gap in the analysis when moving from the randomized projections algorithm of Theorem 4.4.2 to the randomized averaged reflections algorithm of Theorem 4.5.5. In particular, the reflections algorithm assumes metric regularity of  $\Phi$  while the projections algorithm only assumes metric subregularity. Additionally, the analysis of the randomized reflections algorithm requires a “double-application” of metric regularity, by also applying Proposition 2.3.30, leading to a worse convergence rate. It is clear that a metric subregularity assumption on the solution set is insufficient by considering the example where  $S_1$  and  $S_2$  are identical hyperplanes and noting that  $\frac{1}{2}[x + R_{S_1}(R_{S_2}(x))] = \frac{1}{2}[x + R_{S_2}(R_{S_1}(x))] = x$  for all  $x \in \mathbb{E}$ .

One initial attempt could be to apply Theorem 5.2.5. Since the averaged reflection operator of Algorithm 4.5.4 is firmly non-expansive, it is the resolvent of a maximal monotone operator by Proposition 5.1.13. By considering the operator defined by

$$T^{i_1, \dots, i_k}(x) = \begin{cases} \mathbb{R}_+ \text{conv}\{y - R_{S_{i_1}} \dots R_{S_{i_k}}(y)\} & x = \frac{1}{2}[y + R_{S_{i_1}} \dots R_{S_{i_k}}(y)], y \in \mathbb{E} \\ \emptyset & \text{otherwise} \end{cases},$$

where  $\text{conv}(S)$  is the convex hull of  $S$ , it follows that  $\frac{1}{2}[I + R_{S_{i_1}} \dots R_{S_{i_k}}]$  is the resolvent of  $T^{i_1, \dots, i_k}(x)$  and it can be verified that  $T$  is metrically subregular at  $\bar{x}$  for 0 for every  $\bar{x} \in \text{rng} \frac{1}{2}[I + R_{S_{i_1}} \dots R_{S_{i_k}}]$  with the modulus of subregularity being zero. The question that remains is what regularity conditions on

$S_1, \dots, S_m$  are needed to ensure metric subregularity of the related solution mapping in Theorem 5.2.5.

From another perspective, there is some “loss” of information in the analysis when moving from Inequality 4.5.7 to Inequality 4.5.8. Further investigation in this part of the proof may lead to a better convergence rate.

2. Following along the lines of Section 4.5, we could define a general reflection operator  $R_{AT} = 2J_{AT} - I$  and consider an averaged reflections algorithm, similar to that of Algorithm 4.5.4, for solving monotone inclusion problems. By Theorem 5.2.11, we know that performing such an algorithm for  $\lambda \rightarrow \infty$  will converge like the randomized reflections algorithm performed on the solution set. A result on the convergence behavior of this algorithm for arbitrary values of  $\lambda$ —possibly one that incorporates  $\lambda$  into the convergence rate—would be of interest.
3. Practical interest in the proximal point algorithm requires an inexact computation of the proximal step. For the problem of finding a zero of a single maximal monotone operator, conditions on the computational error in each step are given in [102] and [106] to guarantee linear convergence. It would be of interest to generalize this to the problem of finding a common zero of multiple operators to see if a practical rate of convergence can be expressed in terms of the modulus of (sub)regularity and the error.
4. In general, projection algorithms and proximal point algorithms are known to be only weakly convergent. In fact, examples where strong convergence fails are shown in [51] and [14]. On the other hand, a method developed by Haugazeau in [53] was shown to be a strongly convergent variant of alternating projection algorithms. Further, this algorithm was generalized in [11] to include strong convergence for general fixed-point prob-

lems. It would be of interest to see whether the ideas of metric regularity and randomization can provide meaningful linear convergence rates for these types as algorithms while maintaining strong convergence, building upon the results of [106] and [107], among others.

5. Although the ideas presented in Chapter 3 did not emphasize numerical results, some of the methods of Hessian estimation discussed may prove useful in a practical setting, especially where the computational costs of arithmetic operations are low relative to the computational costs of evaluating a function. It would be of interest to further investigate which algorithms could practically take advantage of the idea of this method Hessian estimation.
6. Recent work in [70] focused on using metric regularity to understand the 2-set alternating projection algorithm and the  $m$ -set averaged projection algorithm in the case where the underlying sets are non-convex. Further, [4] focused on using metric regularity to achieve convergence for the proximal point method in the case of non-monotone operators. It would be of interest to see if the idea of randomization in the convex case can be extended to non-convex projections or to the common zero problem for non-monotone operators.

## BIBLIOGRAPHY

- [1] H. Akaike. On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method. *Annals of the Institute of Statistical Mathematics*, 11:1–16, 1959.
- [2] E. Amaldi, P. Belotti, and R. Hauser. Randomized relaxation methods for the maximum feasible subsystem problem. In M. Jünger and V. Kaibel, editors, *Integer Programming and Combinatorial Optimization*, volume 3509 of *Lecture Notes in Computer Science*, pages 249–264, 2005.
- [3] I. Amemiya and T. Ando. Convergence of random products of contractions in Hilbert space. *Acta Sci. Math.*, 26:239–244, 1965.
- [4] F. J. Aragón Artacho, A. L. Dontchev, and M. H. Geoffroy. Convergence of the proximal point method for metrically regular mappings. In *CSVAA 2004—control set-valued analysis and applications*, volume 17 of *ESAIM Proc.*, pages 1–8. EDP Sci., Les Ulis, 2007.
- [5] F.J.A. Artacho and M.H. Geoffroy. Characterization of metric regularity of subdifferentials. *Journal of Convex Analysis*, 15(2):365, 2008.
- [6] H. H. Bauschke. A norm convergence result on random products of relaxed projections in Hilbert space. *Transactions of the American Mathematical Society*, 347(4):1365–1373, 1995.
- [7] H. H. Bauschke and J. M. Borwein. On the convergence of von Neumann’s alternating projection algorithm for two sets. *Set-Valued Analysis*, 1:185–212, 1993.
- [8] H. H. Bauschke and J. M. Borwein. On projection algorithms for solving convex feasibility problems. *SIAM Review*, 38(3):367–426, 1996.
- [9] H.H. Bauschke, J.M. Borwein, and A.S. Lewis. The method of cyclic projections for closed convex sets in Hilbert space. volume 2 of *Recent Developments in Optimization Theory and Nonlinear Analysis: Ams/Imu Special Session on Optimization and Nonlinear Analysis, May 24–26, 1995, Jerusalem, Israel*. American Mathematical Society, 1997.
- [10] H.H. Bauschke, J.M. Borwein, and P. Tseng. Bounded linear regularity, strong CHIP, and CHIP are distinct properties. *Journal of Convex Analysis*, 7(2):395–412, 2000.

- [11] H.H. Bauschke and P.L. Combettes. A weak-to-strong convergence principle for Fejer-monotone methods in Hilbert spaces. *Mathematics of Operations Research*, 26(2):248–264, 2001.
- [12] H.H. Bauschke, P.L. Combettes, and D.R. Luke. Finding best approximation pairs relative to two closed convex sets in Hilbert spaces. *Journal of Approximation Theory*, 127(2):178–192, 2004.
- [13] H.H. Bauschke, P.L. Combettes, and D.R. Luke. A strongly convergent reflection method for finding the projection onto the intersection of two closed convex sets in a Hilbert space. *Journal of Approximation Theory*, 141(1):63–69, 2006.
- [14] H.H. Bauschke, E. Matoušková, and S. Reich. Projection and proximal point methods: convergence results and counterexamples. *Nonlinear Analysis*, 56(5):715–738, 2004.
- [15] A. Beck and M. Teboulle. Convergence rate analysis and error bounds for projection algorithms in convex feasibility problems. *Optimization Methods and Software*, 18(4):377–394, 2003.
- [16] M. Benzi. Gianfranco Cimmino’s contributions to numerical mathematics. In *Seminario di Analisi Matematica, Dipartimento di Matematica dell’Università di Bologna, Ciclo di Conferenze in Ricordo di Gianfranco Cimmino, Marzo-Maggio 2004*, pages 87–109, 2005.
- [17] P. Billingsley. *Probability and Measure*. John Wiley & Sons, 1986.
- [18] J.M. Borwein and A.S. Lewis. *Convex Analysis And Nonlinear Optimization: Theory And Examples*. Springer, 2006.
- [19] K.W. Brodlie. A new direction set method for unconstrained minimization without evaluating derivatives. *Journal of the Institute of Mathematics and Its Applications*, 15:385–396, 1975.
- [20] R. E. Bruck. Random products of contractions in metric and Banach spaces. *Journal of Mathematical Analysis and Applications*, 88:319–332, 1982.
- [21] J.V. Burke and P. Tseng. A unified analysis of Hoffman’s bound via Fenchel duality. *SIAM Journal on Optimization*, 6:265, 1996.
- [22] C. Cenker, H.G. Feichtinger, M. Mayer, H. Steier, and T. Strohmer. New

- variants of the POCS method using affine subspaces of finite codimension, with applications to irregular sampling. In *Proc. SPIE: Visual Communications and Image Processing*, pages 299–310, 1992.
- [23] Y. Censor, G.T. Herman, and M. Jiang. A note on the behavior of the randomized Kaczmarz algorithm of Strohmer and Vershynin. 2009. Preprint available at <http://math.haifa.ac.il/yair/chm-jfaa-rev-080109.pdf>.
  - [24] G. Cimmino. Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari. *La Ricerca Scientifica XVI Series II*, 9(1):326–333, 1938.
  - [25] F.H. Clarke, Y.S. Ledyaev, R.J. Stern, and P.R. Wolenski. *Nonsmooth Analysis and Control Theory*. Springer, 1998.
  - [26] P.L. Combettes. Solving monotone inclusions via compositions of nonexpansive averaged operators. *Optimization*, 53(5):475–504, 2004.
  - [27] R. Cominetti. Metric regularity, tangent sets, and second-order optimality conditions. *Applied Mathematics and Optimization*, 21(1):265–287, 1990.
  - [28] J. W. Demmel. On condition numbers and the distance to the nearest ill-posed problem. *Numerische Mathematik*, 51:251–289, 1987.
  - [29] J. W. Demmel. The probability that a numerical analysis problem is difficult. *Mathematics of Computation*, 50(182):449–480, 1988.
  - [30] F. Deutsch. *Best Approximation in Inner Product Spaces*. Springer-Verlag, New York, 2001.
  - [31] A.V. Dmitruk, A.A. Milyutin, and N.P. Osmolovskii. Lyusternik’s theorem and the theory of extrema. *Russian Mathematical Surveys*, 35(6):11–51, 1980.
  - [32] A.L. Dontchev, A.S. Lewis, and R.T. Rockafellar. The radius of metric regularity. *Transactions of the American Mathematical Society*, 355(2):493–517, 2003.
  - [33] A.L. Dontchev and R.T. Rockafellar. Regularity and conditioning of solution mappings in variational analysis. *Set-Valued Analysis*, 12:79–109, 2004.
  - [34] P. Drineas, R. Kannan, and M.W. Mahoney. Fast Monte Carlo algorithms

- for matrices I: Approximating matrix multiplication. *SIAM Journal on Computing*, 36(1):132–157, 2006.
- [35] P. Drineas, M.W. Mahoney, and S. Muthukrishnan. Sampling algorithms for  $l_2$  regression and applications. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 1127–1136. ACM New York, NY, USA, 2006.
  - [36] J. Dye, M. A. Khamisi, and S. Reich. Random products of contractions in Banach spaces. *Transactions of the American Mathematical Society*, 325(1):87–99, 1991.
  - [37] C. Eckart and G. Young. The approximation of one matrix by another of low rank. *Psychometrika*, 1:211–218, 1936.
  - [38] J. Eckstein. *Splitting Methods for Monotone Operators with Applications to Parallel Optimization*. PhD thesis, Massachusetts Institute of Technology, 1989.
  - [39] Y. Ermoliev, V.I. Norkin, and R. J-B. Wets. The minimization of semicontinuous functions: Mollifier subgradients. *SIAM Journal of Control and Optimization*, 33(1):149–167, 1995.
  - [40] Yu. Ermoliev and R.J.-B. Wets. *Numerical Techniques for Stochastic Optimization*. Springer-Verlag, 1988.
  - [41] H.G. Feichtinger and K. Grochenig. Theory and practice of irregular sampling. In J. Benedetto and M. Frazier, editors, *Wavelets: Mathematics and Applications*, pages 305–363. 1994.
  - [42] A.D. Flaxman, A.T. Kalai, and H.B. McMahan. Online convex optimization in the bandit setting: Gradient descent without a gradient. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 385–394, 2005.
  - [43] A. Frieze, R. Kannan, and S. Vempala. Fast Monte-Carlo algorithms for finding low-rank approximations. *Journal of the Association for Computing Machinery*, 51(6):1025–1041, 2004.
  - [44] A. Galantai. On the rate of convergence of the alternating projection method in finite dimensional spaces. *Journal of Mathematical Analysis and Applications*, 310(1):30–44, 2005.



- [45] M. Gaviano. Some general results on convergence of random search algorithms in minimization problems. In L.C.W. Dixon and G.P. Szegö, editors, *Towards Global Optimisation*, pages 149–157, 1975.
- [46] P. E. Gill, G. H. Golub, W. Murray, and M. A. Saunders. Methods for modifying matrix factorizations. *Mathematics of Computation*, 28(126):505–535, 1974.
- [47] K. Goebel and W.A. Kirk. *Topics in Metric Fixed Point Theory*. Cambridge University Press, 1990.
- [48] G. Golub and C. van Loan. *Matrix Computations*. Johns Hopkins University Press, 1996.
- [49] L.M. Graves. Some mapping theorems. *Duke Mathematical Journal*, 17(1):111–114, 1950.
- [50] L.G. Gubin, B.T. Polyak, and E.V. Raik. The method of projections for finding the common point of convex sets. *U.S.S.R. Computational Mathematics and Mathematical Physics*, 7:1–24, 1967.
- [51] O. Güler. On the convergence of the proximal point algorithm for convex minimization. *SIAM Journal on Control and Optimization*, 29:403, 1991.
- [52] O. Güler, A.J. Hoffman, and U. Rothblum. Approximations to solutions to systems of linear inequalities. *SIAM Journal on Matrix Analysis and Applications*, 16(2):688–696, 1995.
- [53] Y. Haugazeau. *Sur les inéquations variationnelles et la minimisation de fonctionnelles convexes*. PhD thesis, Université de Paris, Paris, France, 1968.
- [54] G.T. Herman and L.B. Meyer. Algebraic reconstruction techniques can be made computationally efficient. *IEEE Transactions on Medical Imaging*, 12(3):600–609, 1993.
- [55] J.B. Hiriart-Urruty and C. Lemaréchal. *Fundamentals of Convex Analysis*. Springer, 2001.
- [56] S.A. Hirstoaga. Iterative selection methods for common fixed point problems. *Journal of Mathematical Analysis and Applications*, 324(2):1020–1035, 2006.

- [57] J.C.K. Ho and L. Tunçel. Reconciliation of various complexity and condition measures for linear programming problems and a generalization of Tardos' theorem. In *Foundations of Computational Mathematics: Proceedings of Smalefest 2000*, pages 93–148, 2002.
- [58] A.J. Hoffman. On approximate solutions of systems of linear inequalities. *J. Res. Nat. Bur. Stand.*, 49:263–265, 1952.
- [59] R. Horn and C. Johnson. *Matrix Analysis*. Cambridge University Press, 1999.
- [60] R.A. Horn and C.R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, 1991.
- [61] A.D. Ioffe. Metric regularity and subdifferential calculus. *Russian Mathematical Surveys*, 55(3):501–558, 2000.
- [62] A.D. Ioffe and J.V. Outrata. On metric and calmness qualification conditions in subdifferential calculus. *Set-Valued Analysis*, 16(2):199–227, 2008.
- [63] S. Kaczmarz. Angenäherte auflösung von systemen linearer gleichungen. *Bull. Int. Acad. Polon. Sci. A*, 355(357), 1937.
- [64] K. Kido. Strong convergence of resolvents of monotone operators in Banach spaces. *Proceedings of the American Mathematical Society*, 103:752–758, 1988.
- [65] K.C. Kiwiel and B. Lopuch. Surrogate projection methods for finding fixed points of firmly nonexpansive mappings. *SIAM Journal on Optimization*, 7(4):1084–1102, 1997.
- [66] D. Klatte and B. Kummer. Optimization methods and stability of inclusions in Banach spaces. *Mathematical Programming*, 117(1-2):305–330, 2009.
- [67] A. Kruger. Stationarity and regularity of set systems. *Pacific Journal of Optimization*, 1:101–126, 2005.
- [68] A.Y. Kruger. About regularity of collections of sets. *Set-Valued Analysis*, 14(2):187–206, 2006.
- [69] N. Lehdili and B. Lemaire. The barycentric proximal method. *Communications on Applied Nonlinear Analysis*, 6:29–47, 1999.

- [70] A.S. Lewis, D.R. Luke, and J. Malick. Local convergence for alternating and averaged nonconvex projections. *Foundations of Computational Mathematics*, to appear. DOI 10.1007/s10208-008-9036-y.
- [71] W. Li. The sharp Lipschitz constants for feasible and optimal solutions of a perturbed linear program. *Linear Algebra and its Applications*, 187:15–40, 1993.
- [72] W. Li. Abadie’s constraint qualification, metric regularity, and error bounds for differentiable convex inequalities. *SIAM Journal of Optimization*, 7:966–978, 1997.
- [73] W. Li and I. Singer. Global error bounds for convex multifunctions and applications. *Mathematics of Operations Research*, 23(2):443–462, 1998.
- [74] D.R. Luke. Relaxed averaged alternating reflections for diffraction imaging. *Inverse Problems*, 21(1):37–50, 2005.
- [75] Z. Q. Luo. New error bounds and their applications to convergence analysis of iterative algorithms. *Mathematical Programming*, 88(2):341–355, 2000.
- [76] Z.Q. Luo and P. Tseng. On the convergence of the coordinate descent method for convex differentiable minimization. *Journal of Optimization Theory and Applications*, 72:7–35, 1992.
- [77] Z.Q. Luo and P. Tseng. Error bounds and convergence analysis of feasible descent methods: a general approach. *Annals of Operations Research*, 46(1):157–178, 1993.
- [78] F.J. Luque. Asymptotic convergence analysis of the proximal point algorithm. *SIAM Journal on Control and Optimization*, 22:277, 1984.
- [79] L.A. Lyusternik. Conditional extrema of functionals. *Mat. Sbornik*, 41(3):390–401, 1934.
- [80] B. Martinet. Regularisation d’inequations variationnelles par approximations successives. *Revue Francaise d’Informatique et de Recherche Operationelle*, 4:154–159, 1970.
- [81] G.J. Minty. Monotone (nonlinear) operators in Hilbert space. *Duke Mathematical Journal*, 29(3):341–346, 1962.

- [82] M.V. Nayakkankuppam and M.L. Overton. Conditioning of semidefinite programs. *Mathematical Programming*, 85(3):525–540, 1999.
- [83] L. Nazareth. Generation of conjugate directions for unconstrained minimization without derivatives. *Mathematics of Computation*, 30(133):115–131, 1976.
- [84] J.A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308, 1965.
- [85] K.F. Ng and X.Y. Zheng. Hoffman’s least error bounds for linear inequalities. *Journal of Global Optimization*, 30:391–403, 2004.
- [86] H.V. Ngai and M. Théra. Metric inequality, subdifferential calculus and applications. *Set-Valued Analysis*, 9(1):187–216, 2001.
- [87] J.-S. Pang. Error bounds in mathematical programming. *Mathematical Programming*, 79:299–332, 1997.
- [88] T. Pennanen. Local convergence of the proximal point algorithm and multiplier methods without monotonicity. *Mathematics of Operations Research*, 27(1):170, 2002.
- [89] G. Pierra. Decomposition through formalization in a product space. *Mathematical Programming*, 28:96–115, 1984.
- [90] B.T. Polyak. Random algorithms for solving convex inequalities. In D. Butnariu, Y. Censor, and S. Reich, editors, *Inherently Parallel Algorithms in Feasibility and Other Applications*. Elsevier, 2001.
- [91] M.J.D. Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal*, 7:155–162, 1964.
- [92] M.J.D. Powell. On search directions for minimization algorithms. *Mathematical Programming*, 4:193–201, 1973.
- [93] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*. Springer-Verlag, 2007.
- [94] G. Rappl. On linear convergence of a class of random search algorithms. *ZAMM - Journal of Applied Mathematics and Mechanics*, 69(1):37–45, 1989.

- [95] S. Reich. A limit theorem for projections. *Linear and Multilinear Algebra*, 13(3):281–290, 1983.
- [96] J. Renegar. Perturbation theory for linear programming. *Mathematical Programming*, 65:73–91, 1994.
- [97] J. Renegar. Incorporating conditions measures into the complexity theory of linear programming. *SIAM Journal on Optimization*, 5(3):506–524, 1995.
- [98] J. Renegar. Linear programming, complexity theory and elementary functional analysis. *Mathematical Programming*, 70:279–351, 1995.
- [99] S.M. Robinson. An application of error bounds for convex programming in a linear space. *SIAM Journal on Control*, 13:271–273, 1975.
- [100] S.M. Robinson. Regularity and stability for convex multivalued functions. *Mathematics of Operations Research*, 1(2):130–143, 1976.
- [101] R.T. Rockafellar. On the maximal monotonicity of subdifferential mappings. *Pacific Journal of Mathematics*, 33(1):209–216, 1970.
- [102] R.T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM Journal on Control and Optimization*, 14(5):877–898, 1976.
- [103] R.T. Rockafellar and R.J.-B. Wets. *Variational Analysis*. Springer, Berlin, 1998.
- [104] A. Shapiro, T. Homem-de Mello, and J. Kim. Conditioning of convex piecewise linear stochastic programs. *Mathematical Programming*, 94(1):1–19, 2002.
- [105] F.J. Solis and R. J-B. Wets. Minimization by random search techniques. *Mathematics of Operations Research*, 6(1):19–30, 1981.
- [106] M.V. Solodov and B.F. Svaiter. A hybrid projection-proximal point algorithm. *Journal of Convex Analysis*, 6:59–70, 1999.
- [107] M.V. Solodov and B.F. Svaiter. Forcing strong convergence of proximal point iterations in a Hilbert space. *Mathematical Programming*, 87(1):189–202, 2000.

- [108] J.C. Spall. *Introduction to Stochastic Search and Optimization: Estimation, Simulation and Control*. John Wiley & Sons, New York, 2003.
- [109] T. Strohmer and R. Vershynin. A randomized Kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2):262–278, 2009.
- [110] M.J. Todd, L. Tunçel, and Y. Ye. Characterizations, bounds, and probabilistic analysis of two complexity measures for linear programming problems. *Mathematical Programming*, 90(1):59–69, 2001.
- [111] C. Ursescu. Multifunctions with convex closed graph. *Czechoslovak Math. J.*, 25(100):438–441, 1975.
- [112] M.H. Wright. Ill-conditioning and computational error in interior methods for nonlinear programming. *SIAM Journal on Optimization*, 9(1):84–111, 1998.
- [113] W.I. Zangwill. Minimizing a function without calculating derivatives. *The Computer Journal*, 10(3):293–296, 1967.
- [114] S. Zhang. Global error bounds for convex conic problems. *SIAM Journal on Optimization*, 10(3), 2000.
- [115] X.Y. Zheng and K.F. Ng. Metric subregularity and constraint qualifications for convex generalized equations in Banach spaces. *SIAM Journal of Optimization*, 18(2):437, 2008.
- [116] T. Zolezzi. On the distance theorem in quadratic optimization. *Journal of Convex Analysis*, 9(2):693–700, 2002.